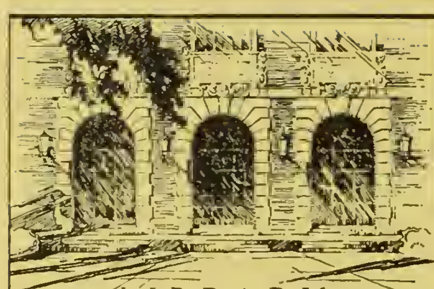


Report No. 273

AN ANALYSIS OF THE REGRESSION MODEL FOR NONSATURATING LOGIC CIRCUIT ANALYSIS

by

George Donald Wood

May 10, 1968

DEPARTMENT OF COMPUTER SCIENCE · UNIVERSITY OF ILLINOIS · URBANA, ILLINOIS

Report No. 273

AN ANALYSIS OF THE REGRESSION MODEL FOR NONSATURATING LOGIC CIRCUIT ANALYSIS*

by

George Donald Wood

May 10, 1968

Department of Computer Science
University of Illinois
Urbana, Illinois 61801

*

* This work was supported in part by the Advanced Research Projects Agency as administered by the Rome Air Development Center under Contract No. US AF 30(602)4144 and submitted in partial fulfillment of the requirements for the degree of Master of Science in Electrical Engineering, June, 1968.

ACKNOWLEDGEMENT

The author acknowledges with pleasure the assistance and encouragement of Professor Richard M. Brown, whose many helpful comments and suggestions contributed greatly to this thesis project. Also, Professor Thomas A. Murrell, in whose researches the problem treated here originated, contributed much time and many useful suggestions during a number of thoughtful discussion periods.

The author would also like to express his sincere gratitude to the Department of Computer Science of the University of Illinois and to the ILLIAC IV project for providing financial support for a portion of this project.

Many thanks are also due to Mrs. Mildred Pape for her diligent and careful typing of the manuscript.

TABLE OF CONTENTS

Page

1. INTRODUCTION 1

2. THE DEVELOPMENT OF THE REGRESSION MODEL AND ITS RELATIONSHIP TO CONVENTIONAL CIRCUIT MODELING 7

   2.1 Basic Modeling Elements 7

   2.2 Small Signal Parameter Models 9

   2.3 Piecewise Linear Model Construction 13

   2.4 Ideal Diode Model Construction 16

   2.5 The Regression Model 21

3. COMPARISON OF TWO COMPUTER PROGRAM CIRCUIT ANALYSIS METHODS 30

   3.1 Discussion of a Basic Circuit Analysis Problem . . . 30

   3.2 Solution of the Current Convergence Problem - Matrix Partition Method 30

   3.3 Solution of the Current Convergence Problem - Regression Model Method 39

   3.4 Cutoff Conditions and the Regression Model 49

   3.5 Extension of the Regression Model 53

   3.6 Conclusion 54

4. APPLICATION OF NLAP TO LOGIC CIRCUIT ANALYSIS 63

   4.1 Introduction 63

   4.2 Preparation of NLAP Input Data Deck 63

   4.3 Solution Control Codes 67

   4.4 Solution Output and Identification 68

   4.5 Conclusion 69

REFERENCES 72

Page

APPENDIX

A. A NETWORK ANALYSIS PROCEDURE 75

1. Introduction 75
2. Matrix Equation Formulation 75

B. REGRESSION MODEL SUBROUTINE 84

1. Subroutine Iterate Listing 84
2. Comments on Program Changes 92

C. LINEAR REGRESSION PROGRAM 93

1. Linear Regression Program Listing 93
2. Typical Output 99
3. Instructions for Preparing Linear Regression Program Control Cards 103
4. Typical Input Data Deck 106
5. Linear Regression Input Data Transformation Program 108

D. REGRESSION MODEL EFFECTIVENESS PROGRAM 109

1. Regression Model Effectiveness Program Listing 109
2. Typical Output 110
3. Program Comments 112
4. Typical Input Data Deck 114

E. NLAP OUTPUT SUBROUTINE LISTING 115

LIST OF FIGURES

Figure		Page
1.	Two Forms of Emitter Coupled Logic (ECL)	2
2.	Emitter Coupled Logic Transfer Characteristic	3
3.	Dependent Sources and Their Voltage-Current Characteristics	8
4.	Computer Program for Calculating $I_C = f(I_B)$	14
5.	Piecewise Linear Approximation of a Function	15
6.	Cl Transistor Common Emitter Output Characteristic .	17
7.	Stepwise Development of an NPN Common Emitter Transistor Model	19
8.	Comparison of Hybrid and ECAP Transistor Models . . .	22
9.	Regression Model with Equations	24
10.	Cl Transistor Base to Emitter Voltage Versus Collector Current	28
11.	Cl Transistor Current Gain Versus Collector Current	29
12A.	Oven Heater Driving Amplifier Schematic Diagram . . .	31
12B.	Oven Heater Driving Amplifier Equivalent Circuit . .	32
13.	Dependent Current Generator Matrix	34
14.	Independent Voltage Source Matrix	35
15.	Coefficient Matrix Element Values	37
16.	Impedance Matrix Element Values	38
17.	NLAP Common Emitter Test Circuit	41
18.	ECAP Standard Branch	42
19.	Nodal Conductance Matrix with BETA Explicitly Shown .	47

Figure | Page

20. Nodal Equivalent Current Vector with BETA and V_{BE} Explicitly Shown 48

21. Flow Chart of Iterative Regression Model Solution Method 55

22. DC Solution Matrix Obtained Prior to Iteration . . 56

23. NL Solution Matrix Obtained After Completion of Iteration 57

24. Nodal Conductance Matrix Calculations for BETA = 50 58

25. Nodal Current Vector Calculations for BETA = 50 and V_{BE} = -1.25 Volts 59

26. OR Emitter Follower Voltage Offset Characteristic . 61

27. OR Emitter Follower Schematic Diagram 62

28. NLAP Calling Sequence 64

29. Typical NLAP Input Data 66

30. Typical NLAP 133 Column Solution Printed Output . . 71

A1. An Equivalent Network 76

A2. Branch Level E, JG, V, and I Matrices 77

A3. Branch Level Z and A Matrices 78

A4. Network Analysis Program 82

A5. Output of Network Analysis Test Program 83

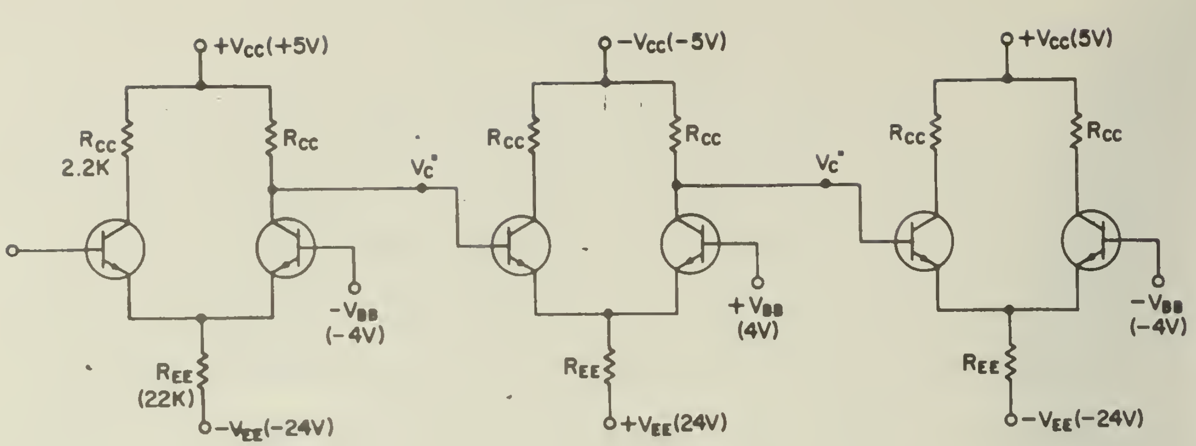
LIST OF TABLES

Table		Page
1.	Analysis of Regression Equation for BETA, C_1 $V_{CE} = 3.0\ V_{DC}$	51
2.	Analysis of Regression Equation for V_{BE}, C_1 $V_{CE} = 3.0\ V_{DC}$	52
3.	Summary of NLAP Output Variable Storage Arrays . . .	70

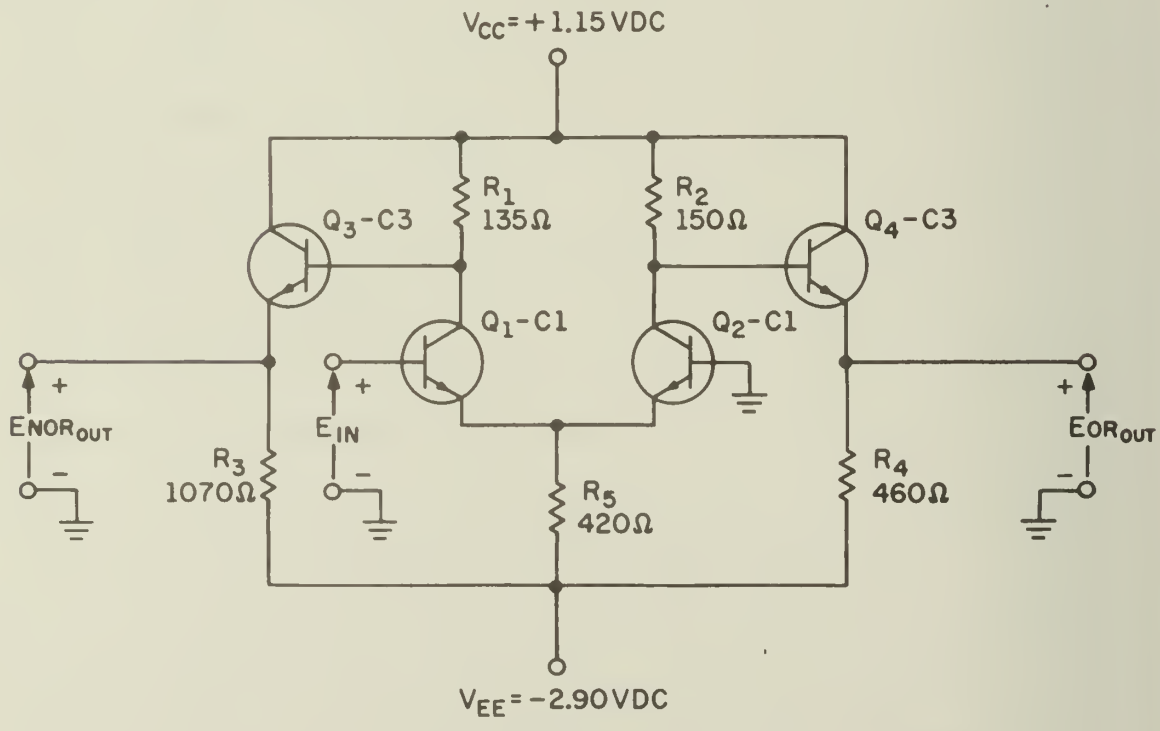
1. INTRODUCTION

Many approaches to the computer aided design problem[1] are being studied. To date these efforts have not resulted in the general availability of automatic circuit design programs[2]. If, however, one subdivides the electronic circuit design process somewhat, one of the principle facets which emerges is the need for a through analysis capability. With such a tool available to the designer, electronic system response may be simulated over the entire space of component variations and operating conditions.

One of the fundamental steps in designing electronic circuits is the choice of the "static d.c." or quiescent operating point. Its optimization is often an iterative process. An example, from logic circuit design, occurs when designing emitter coupled logic circuits[3] such as those of Figure 1. The typical design process requires that a number of interdependencies be met in order to obtain the nonsaturating output characteristics of Figure 2. The process begins with the selection of the voltage levels representing each binary state. The on and off state base-to-emitter voltages may be determined by superimposing[4] the source line upon a set of the transistor's input characteristic curves. The output levels are next determined for various loadings, such as the next stage of Figure 1A, or the output level restoring emitter followers of Figure 1B, using the transistor's output characteristic curve family. This procedure is then repeated for varying input and output conditions to insure the convergence of the logic levels[5] to the desired voltage levels.



A) Complementary Direct Coupled ECL



B) Emitter Follower Buffered ECL

Figure 1. Two Forms of Emitter Coupled Logic (ECL)

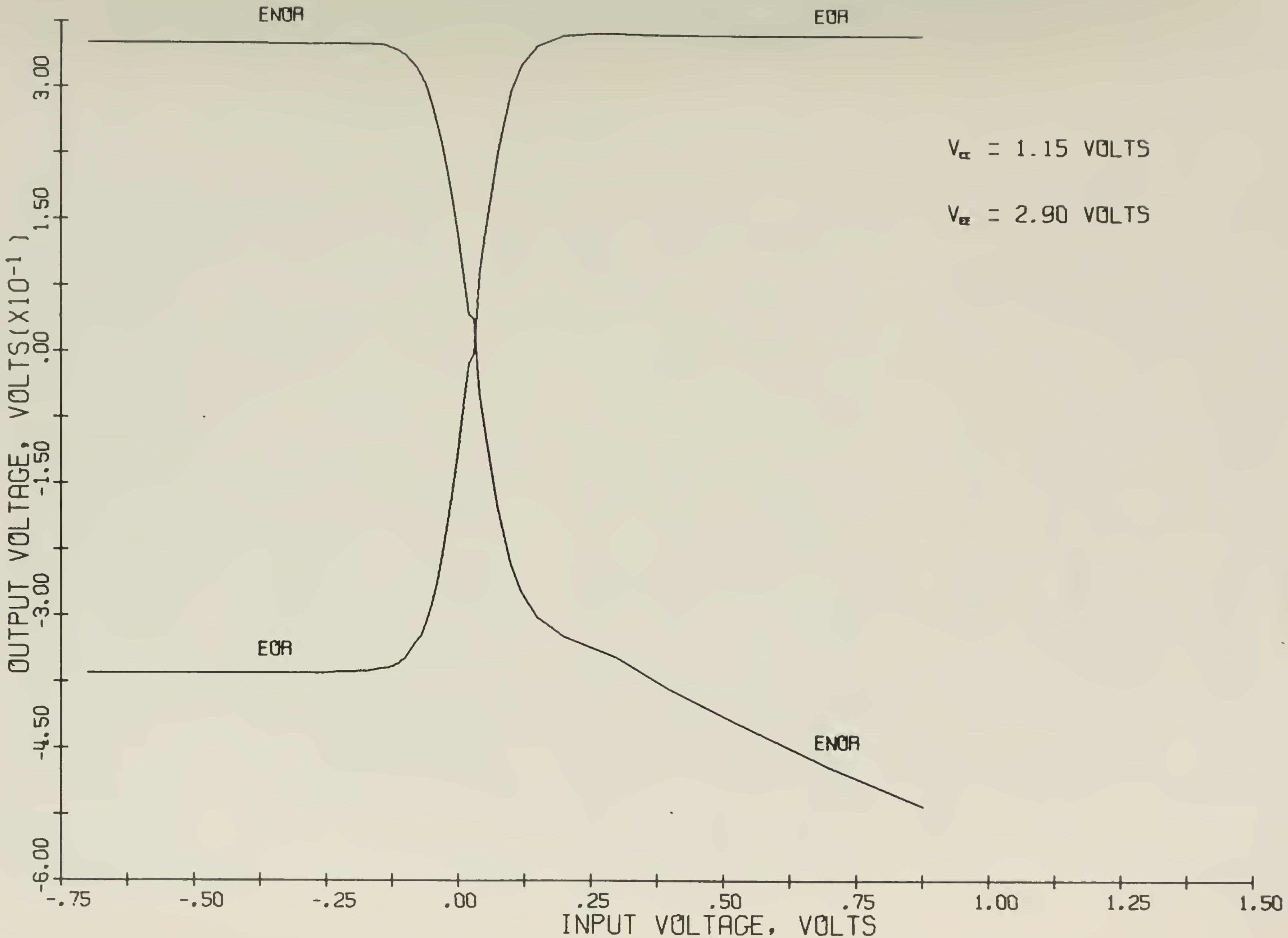


Figure 2. Emitter Coupled Logic Transfer Characteristic

The amount of calculation required for effective design of an emitter coupled logic block can be substantial due to the numerous iterations which may be needed to prove satisfactory operation under varying tolerance conditions. The point being made here is that the circuit designer could save a significant amount of computation time by employing a flexible computer program whose output could be, for example, the emitter coupled logic circuit's output voltage transfer characteristic.

The above "static" design discussion employed graphical analysis which has been a standard circuit design tool[6] for many years. Introductory electronic circuit texts introduce this approach as being the best way to handle nonlinear circuit elements and indeed it is for simple exploratory analysis. However, for computer-aided design, such as is possible with a parameter variation program, it is necessary to enter the transistor's characteristics in the form of equations. These equations are known as the transistor modeling equations and may take many forms.

Some of the models currently used are: Ebers-Moll, Piecewise Linear, Admittance or Y Parameter, Hybrid or H Parameter, Z matrix, and the M Parameter Model. Each model evolved for the purpose of simulating some portion of a transistor's range of operation. No analysis or simulation program can be effective unless the transistor model used accurately represents the transistor's voltage-current behavior throughout the region of operation in the simulated circuit. For this reason, the basis of each of the above models will

be discussed in section 2 in order that they may be compared with a new model[7] related to the Z matrix model but including the additional constraints:

$$\mathrm{BETA} = A_0 + A_1 \left(\ell n \frac{I_C}{I_0}\right) + A_2 \left(\ell n \frac{I_C}{I_0}\right)^2$$

$$V_{BE} = B_0 + A_1 \left(\ell n \frac{I_C}{I_0}\right) + A_2 \left(\ell n \frac{I_C}{I_0}\right)^2$$

These additional constraints are formed by the Linear Regression program of Appendix B whose input data points are taken from measurements or from manufacturer's data sheets. This Regression Model thereby enables the designer to employ linear equation solution methods to obtain a nonlinear solution to his circuit problem. Thus the quantities BETA and V_{BE} are no longer restricted to constant values as is the usual case.

The usefulness of the Regression Model rests upon the ease of its application and upon the accuracy of its predictive capability. In an effort to satisfy the first goal, the Regression Model subroutine written by Guth[8] was converted from IBM 1620 compatible language to FORTRAN IV for use on the IBM 360 Model 75 as a part of this thesis project.

In the process of conversion a new version of ECAP[9] became available which provided true worst case solutions. This new version of ECAP was modified to work with an updated version of Guth's subroutine, a listing of which is given in Appendix B of this thesis.

The complete program is now called NLAP (Nonlinear Analysis Program) and is contained in the program library at the Department of Computer Science, University of Illinois. The NLAP calling sequence is given in Appendix E, together with the input and output of a test program.

The final goal posed for this thesis was to evaluate the possibility of extending the useful range of the Regression Model from the normal active region to include operation in the cutoff region as is encountered in emitter coupled logic circuit operation. The evaluation of this extension, together with a discussion of the IBM 360 ECAP network solution method, will be presented in section 3.

The application of NLAP to the solution of transistor logic circuits is discussed in section 4. The required topological codes are given for the various types of branch level statements. Two illustrative examples are presented as a further guide. It is recommended that the user consult sections 3.4 through 3.6 before undertaking use of the NL solution option discussed in section 4.3.

2. THE DEVELOPMENT OF THE REGRESSION MODEL AND ITS RELATIONSHIP TO CONVENTIONAL CIRCUIT MODELING

2.1 Basic Modeling Elements

The basic components for an electronic model are the idealized pure resistance (R), pure inductance (L), and pure capacitance (C) elements of linear and reciprocal network theory. Their elemental voltage-current relationships are simply Ohm's law. To form a compact set of elements, it is necessary to add the controlled source for modeling of the amplification process and the ideal junction diode for simulating variable amplitude, unilateral transmission.

Dependent current and voltage controlled sources are of four[10] possible types; voltage controlled voltage, voltage controlled current, current controlled voltage, and current controlled current. The internal impedances of these sources are identical with their independent counterparts of linear network theory. The direction of output current flow, or of output voltage polarity, is dependent upon the driving source and may be of either polarity. The two port voltage-current relationships of each controlled source type, together with their schematics, are summarized by Figure 3.

The ideal junction diode has the voltage-current relationship:

$$i = i_0 \left(e^{qV/MKT} - 1\right).$$

This equation derives from a solution of the one-dimensional diffusion equation.[11] The factor M in the denominator of the exponent facilitates adjusting the theoretical V-I relationship to match that observed physically. Use of this ideal diode equation, together with the other

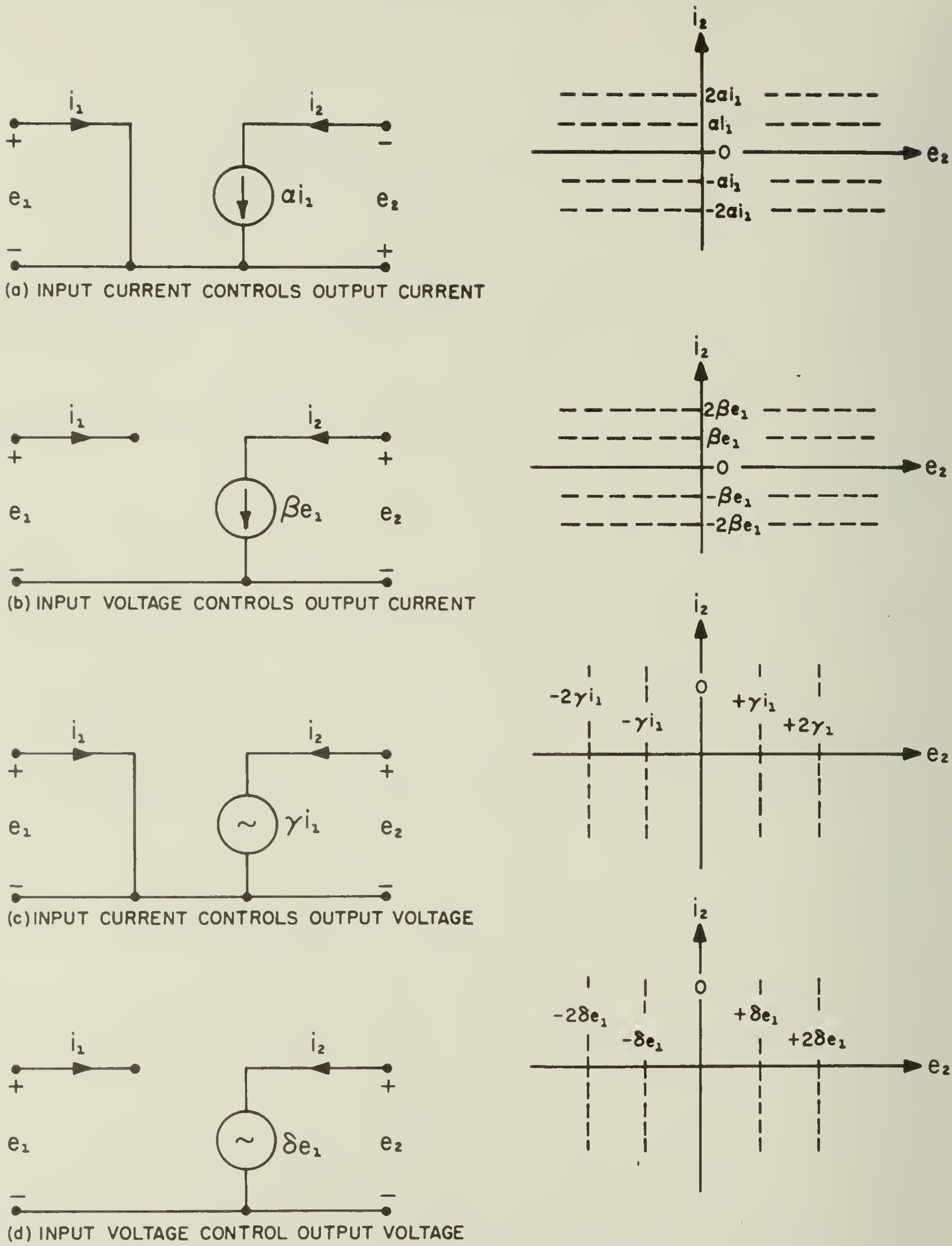


Figure 3. Dependent Sources and Their Voltage-Current Characteristics

four basic modeling elements, plus an independent current or voltage source element[12], are sufficient for most modeling work whether the circuit model be derived from basic physics or from the circuit "black box" approach.

2.2 Small Signal Parameter Models

A general network or "black box" may be described in terms of its two port parameters. At each port the voltage and input current are measured, providing four variables for network description. Any two of these quantities form a basis[13] for the solution space of the network and, therefore, they completely characterize the network's behavior. The two port parameters may be established from a graphical plot by following the definitions established during their derivation.

One pair of general relationships between the two port voltages and current is:

$$e_1 = f\,(i_1, i_2)$$

$$e_2 = g\,(i_1, i_2)$$

These general functional relationships will in general be nonlinear. This creates the need for their transformation into linear functions for use in linear network analysis. Expanding in a Taylor series, with two independent variables, obtains:

$$e_1 (i_1, i_2) = e_1 (I_1, I_2) + \left.\frac{\partial e_1}{\partial i_1}\right|_{\substack{i_1 = I_1 \\ i_2 = I_2}} (i_1 - I_1) + \left.\frac{\partial e_1}{\partial i_2}\right|_{\substack{i_1 = I_1 \\ i_2 = I_2}} (i_2 - I_2)$$

+ higher order terms

$$e_2 (i_1, i_2) = e_2 (I_1, I_2) + \left.\frac{\partial e_2}{\partial i_1}\right|_{\substack{i_1 = I_1 \\ i_2 = I_2}} (i_1 - I_1) + \left.\frac{\partial e_2}{\partial i_1}\right|_{\substack{i_1 = I_1 \\ i_2 = I_2}} (i_2 - I_2)$$

+ higher order terms

provided $e_1 (i_1, i_2)$ and $e_2 (i_1, i_2)$ are analytic functions. The crucial point is that there are two ways for these Taylor expansions to reduce to linear functions. First, if the basic functional relationships are linear, then the value of all derivatives of higher order than one are zero. Second, if the operating point $Q (i_1, i_2)$ is selected such that the deviations $(i_1 - I_1)$ and $(i_2 - I_2)$ are small (thereby restricting the validity of related circuit calculations to "small signal") then the higher order Taylor series terms, $(i_j - I_j)^n$, become sufficiently small so as to become negligible.

Several changes of variables, such as:

$$e_j (i_1, i_2) = e_j (I_1, I_2) + \Delta e_j$$

yields the equations:

$$\Delta e_i = \frac{\partial e_1}{\partial i_1}\Bigg|_{\substack{i_1 = I_1 \\ i_2 = I_2}} \Delta i_1 + \frac{\partial e_1}{\partial e_2}\Bigg|_{\substack{i_1 = I_1 \\ i_2 = I_2}} \Delta i_2$$

$$\Delta e_2 = \frac{\partial e_2}{\partial i_1}\Bigg|_{\substack{i_1 = I_1 \\ i_2 = I_2}} \Delta i_1 + \frac{\partial 3_2}{\partial i_2}\Bigg|_{\substack{i_1 = I_1 \\ i_2 = I_2}} \Delta i_2$$

in which the quantities e_j's and i_j's are incremental by definition with one exception; the case where the basic function is sufficiently linear for all the higher partial derivatives to be at least close to zero.

The partial derivatives $\frac{\partial e_j}{\partial i_k}$ each have the dimensions of impedance. With this in mind, the incremental equations may be written in matrix form where the definitions of Z_{ij} are obvious by inspection.

$$\begin{bmatrix} \Delta e_1 \\ \\ \Delta e_2 \end{bmatrix} = \begin{bmatrix} z_{11} & z_{12} \\ \\ z_{21} & z_{22} \end{bmatrix} * \begin{bmatrix} \Delta i_1 \\ \\ \Delta i_2 \end{bmatrix}$$

Now we arrive at the fortunate result where it is not necessary to assume[14] small signal conditions in order to obtain a valid linear model of a transistor using "small signal Z parameters". However, it is necessary to locate those portions of the operating

regions for each transistor where all four z_{ij}'s are sufficiently linear before this is a valid model.

There are four[15] other sets of "small signal" parameters:

A) The Y, or admittance parameters:

$$\begin{bmatrix} i_1 \\ \\ i_2 \end{bmatrix} = \begin{bmatrix} y_{11} & y_{12} \\ \\ y_{21} & y_{22} \end{bmatrix} * \begin{bmatrix} e_1 \\ \\ e_2 \end{bmatrix}$$

B) The H, or hybrid parameters:

$$\begin{bmatrix} e_1 \\ \\ i_2 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ \\ h_{21} & h_{22} \end{bmatrix} * \begin{bmatrix} i_1 \\ \\ e_2 \end{bmatrix}$$

C) The A, or transfer parameters:

$$\begin{bmatrix} e_1 \\ \\ i_1 \end{bmatrix} = \begin{bmatrix} A & B \\ \\ C & D \end{bmatrix} * \begin{bmatrix} e_2 \\ \\ I_2 \end{bmatrix}$$

D) The M, or inverse hybrid parameters:

$$\begin{bmatrix} i_1 \\ \\ e_2 \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} \\ \\ m_{21} & m_{22} \end{bmatrix} * \begin{bmatrix} e_1 \\ \\ i_2 \end{bmatrix} .$$

Each of the above coefficient matrices, in its linear region may be arranged to represent common emitter, common base, or

common collector orientations through a series of well-known transformations[16]. In this respect this matrix representation is very useful for model construction.

In summary, the so-called small signal parameter models may form useful quiescent models in the regions where all four coefficients of each coefficient matrix are linear. However, lacking sufficient data on which to justify their use, the systems analyst should use a distinctly nonlinear model such as a Piecewise Linear Model, Ideal Diode Model, or Regression Model which is valid over the range of interest.

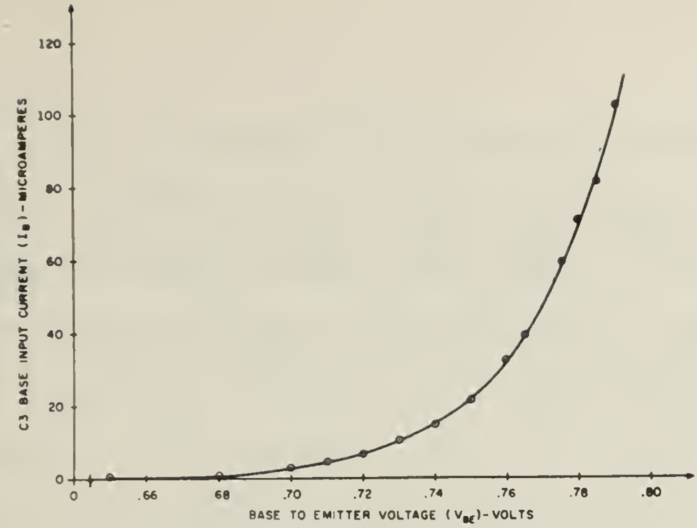
2.3 Piecewise Linear Model Construction

Another class of models may be generated using piecewise linear techniques[17]. The Piecewise Linear Models are also valid for a wide range of operating points provided a sufficient number of segments are employed. An example of a Piecewise Linear Model of a forward base-emitter diode coded in the Jovial subset of ALGOL is shown in Figure 4. In a computer program, such a model consists of logical statements defining the values at which the individual segments should be switched in and out of the model's network.

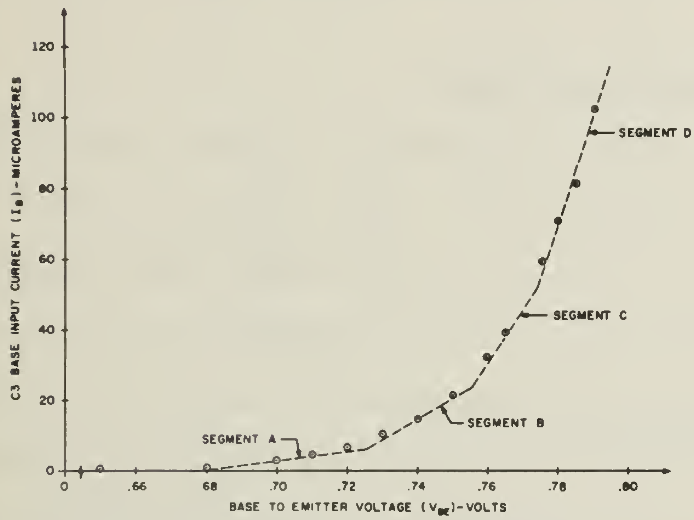
Construction of a Piecewise Linear Model begins with a graph of the objective function (Figure 5). A series of straight line segments are then constructed which, when appropriately summed, form the required objective function within the desired accuracy. An example Piecewise Linear Model for the I_B versus V_{BE} input characteristic of an NPN switching transistor is presented in Figure 5B. Several other

```
*  FLOAT;
*  READ E17;
*  Q5RS = 0.60;
*  ZO = 40;
*  FORMAT HD5,S20,C*CHARACTERISTICS OF TRANSISTOR Q5*//;
*  FORMAT HD5U,C*     IB      IC      HFE      RC      VSAT
*       VCE      VOUT      ES*/;
*  PRINT HD5;
*  PRINT HD5U;
*  FOR IB5 = 0.005, 0.005, 0.060;
*  BEGIN
*  IC5MAX = ((E17/ZO)-0.45)/13.09524;
*  IF IB5 GQ 0 AND IB5 LS 0.007;
*   IC5 = 52.5*IB5;
*  IF IB5 GQ 0.007 AND IB5 LS 0.014;
*   IC5 = 0.18 + 26.66667* 135;
*  IF IB5 GQ 0.014 AND IB5 LS 0.0325;
*  IC5 = 0.3 + 17.71429*IB5;
*  IF IB5 GQ 0.0325 and IB5 LQ IB5MAX;
*   IC5 = 0.45 + 13.09524*IB5;
*  Z1 = (E17/IC5)-ZO;
*  HFE = IC5/IB5;
*  VSAT = IC5*Q5RS;
*  VCE = IC5*Z1;
*  VOUT = E17 - IC5*(Z1+Q5RS);
*  ES = VSAT + VCE + VOUT;
*  FORMAT Q,F7.4,F10.3, F9.1,2F9.s,Sf8.2,F9.2;
*  PRINT Q,IB5,IC5,HFE,Z1,VSAT,VCE,VOUT,ES;
*  END
*  FORMAT QS,C* */,H8,F12.4,C* *//;
*  PRINT QS,8H(IB5MAX =),IB5MAX;
* PRINT COMPLETE
```

Figure 4. Computer Program for Calculating $I_C = f(I_B)$



A) C3 Emitter Diode Input Characteristic



B) Piecewise Linear Approximation

Segment	Equation
A	$V_{BE_A} = 0.680 + 192.85 * I_B$
B	$V_{BE_B} = 0.727 + 591.39 * I_B$
C	$V_{BE_C} = 0.755 + 1569.23 * I_B$
D	$V_{BE_D} = 0.774 + 3153.84 * I_B$

C) Piecewise Linear Equations

Figure 5. Piecewise Linear Approximation of a Function

approaches for generating piecewise linear models are given in Anner[18], which is the most comprehensive reference on the subject. One constraint on the use of Piecewise Linear Models is that they must be entered into a computer program in segments whose equations must be switched in or out of the model as a function of V_{BE} by conditional program logic statements, such as the "IF statement" of FORTRAN IV[19], in order for the total functional value $I_B = f(V_{BE})$ to be correct.

2.4 Ideal Diode Model Construction

Suppose a problem is given whose solution requires rather exact simulation of a transistor's output characteristic such as is depicted in Figure 6. This set of curves shows a set of measured, static characteristics of an Emitter-Coupled-Logic nanosecond switching transistor. Contrasting these with the voltage-current relationships of the controlled sources of Figure 3 suggests that with the application of appropriate mathematical nonnegativity[20] restrictions one could form a suitable model.

An effective way of realizing the nonnegativity restrictions is by adding ideal junction diode elements to the model. Thus, by employing the diode model in appropriate series and parallel configurations, the controlled sources for quadrant V-I characteristic may be transformed into the desired form. One possible algorithm using the current and voltage conventions of Figure 7A is:

A) Shunt the controlled current source with an ideal resistive element, R_C, to provide a modified set of

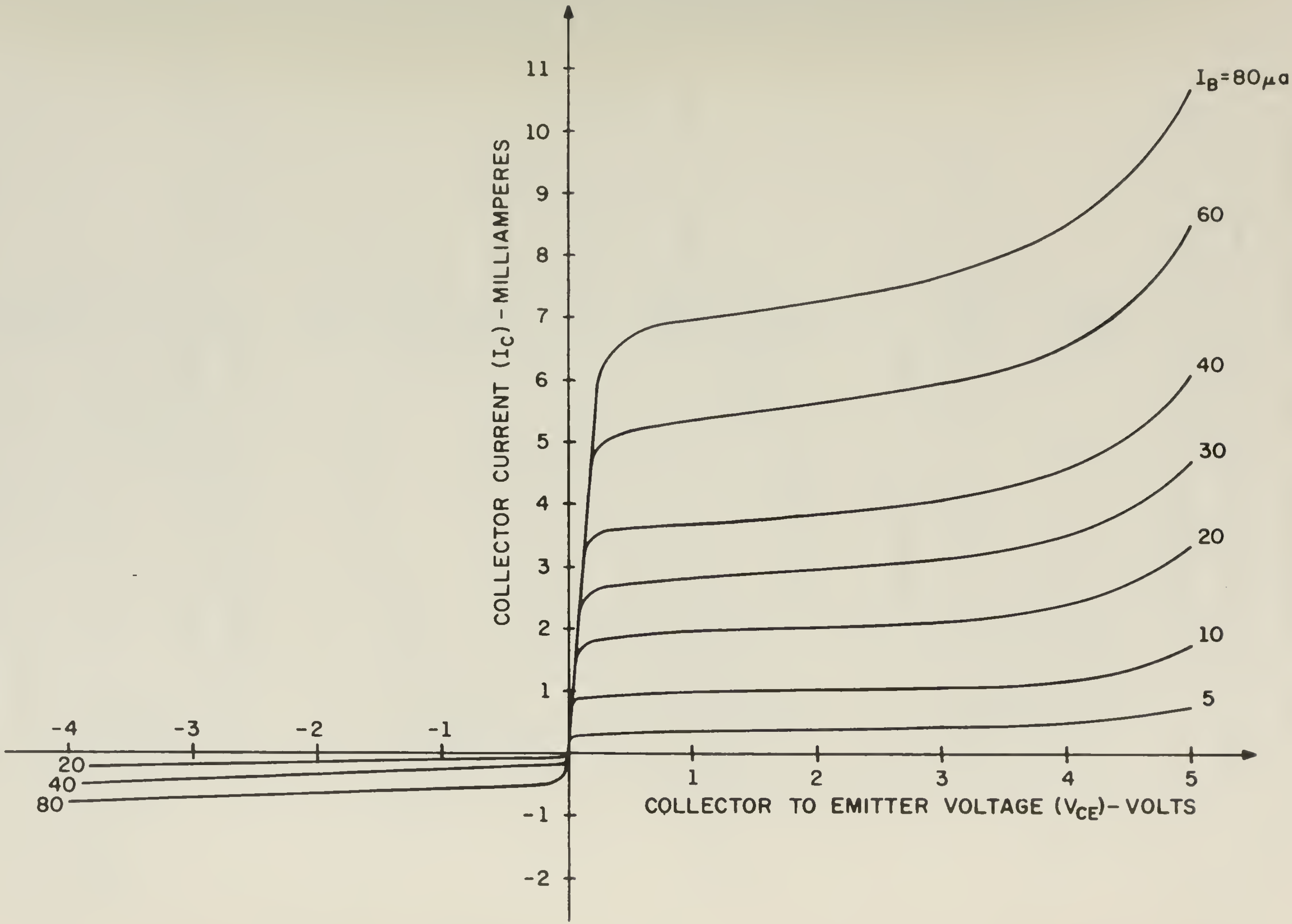


Figure 6. C1 Transistor Common Emitter Output Characteristic

V-I characteristics which have a linearly increasing positive slope of 1/R mhos and whose ordinate values at $V_{CE} = 0$ are unaltered as shown in Figure 7B. Thus:

$$i_C = i_1 + i_R$$

where i_R is the resistive current having a linear dependency upon the output voltage V_{CB} since

$$i_R = V_{CB}/R_C.$$

B) Inhibit the output current i_C by placing an ideal junction diode element in series with the input branch such that current i_E now flows through the ideal junction diode unidirectionally. The output current i_C becomes:

$$i_C = \partial * i_E + i_R$$

and the quasi-model has the voltage current characteristic of Figure 7C, where ∂ is the current gain of the model.

C) Shunt the terminals, across which the potential V_{CB} appears by placing the ideal junction element so

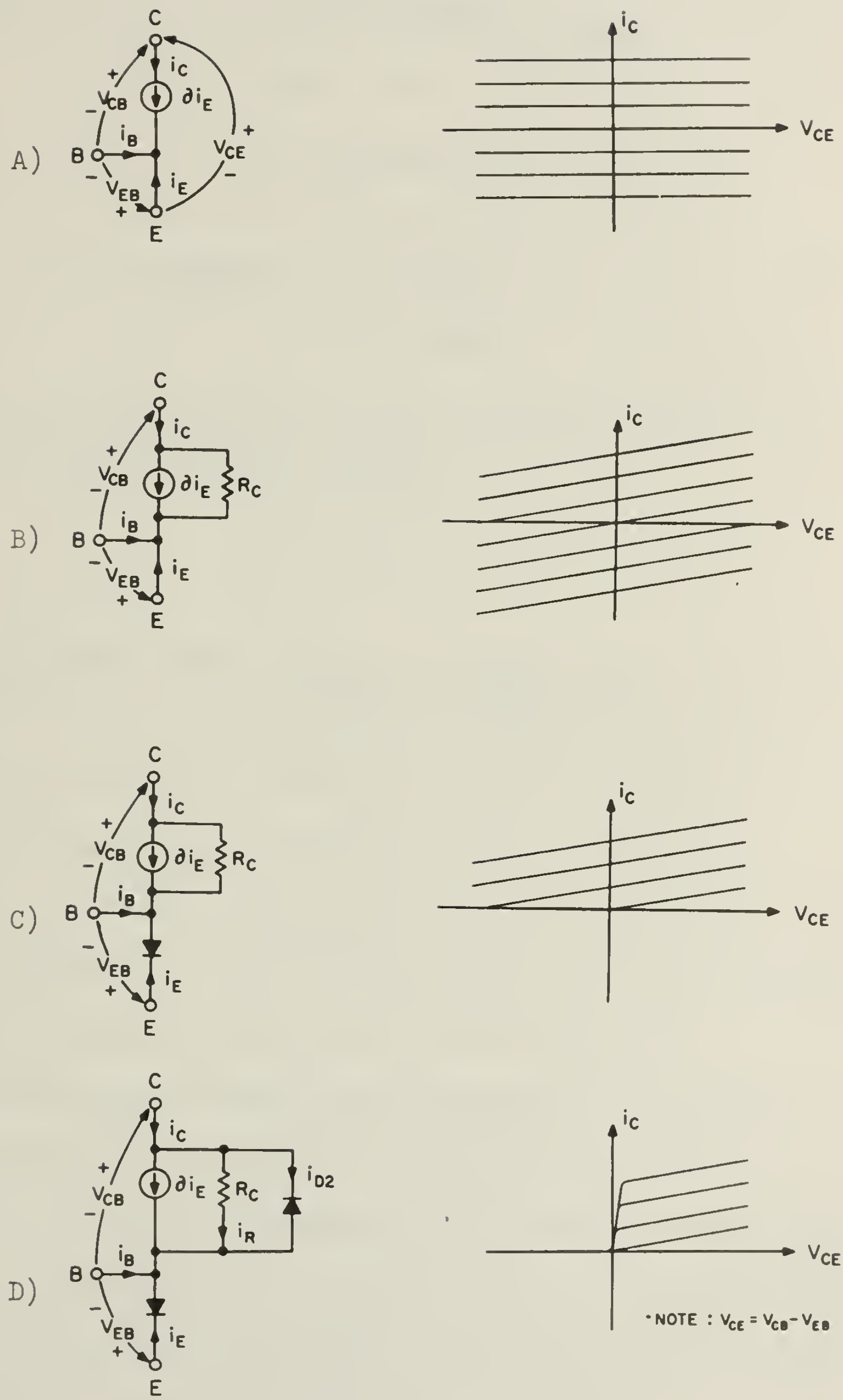


Figure 7. Step-wise Development of an NPN Common Emitter Transistor Model

as to inhibit V_{CB} from becoming negative. The output current is by KCL:

$$i_C = \alpha_F I_{01}\,(e^{-qVEB/Mkt}-1) + V_{CB}/R_C - I_{02}\,(e^{-qVEB/Mkt}-1)$$

while the input current is:

$$i_E = I_{01}\,(e^{-qVEB/Mkt}-1).$$

By KCL and KVL these equations may be couched in normal common emitter variables; I_B instead of I_E and V_{CE} instead of V_{CB} with the result:

$$i_C = \frac{\alpha_F}{1-\alpha_F}\, I_B + \frac{I_{CO}}{1-\alpha_F}\,(e^{-q(V_{CE} - V_{BE})/Mkt}-1)$$

$$i_B = -I_{ES}\,(1-\alpha_F)\,(e^{-qV_{BE}/Mkt}-1) + I_{CS}\,(e^{-q(V_{CE}-V_{BE})Mkt}-1) - V_{CB}/R_C$$

which are the equations of the curves in Figure 7D.

These relations, according to Gibbons[21], provide a representation of transistor output characteristics which is sufficiently accurate that it falls within the typical manufacturing spread for a given transistor type over most of the normal active region. Also, one finds

upon comparing the Ebers Moll Model equations[22] with those of paragraph C above, that there is good agreement in the forward active region.

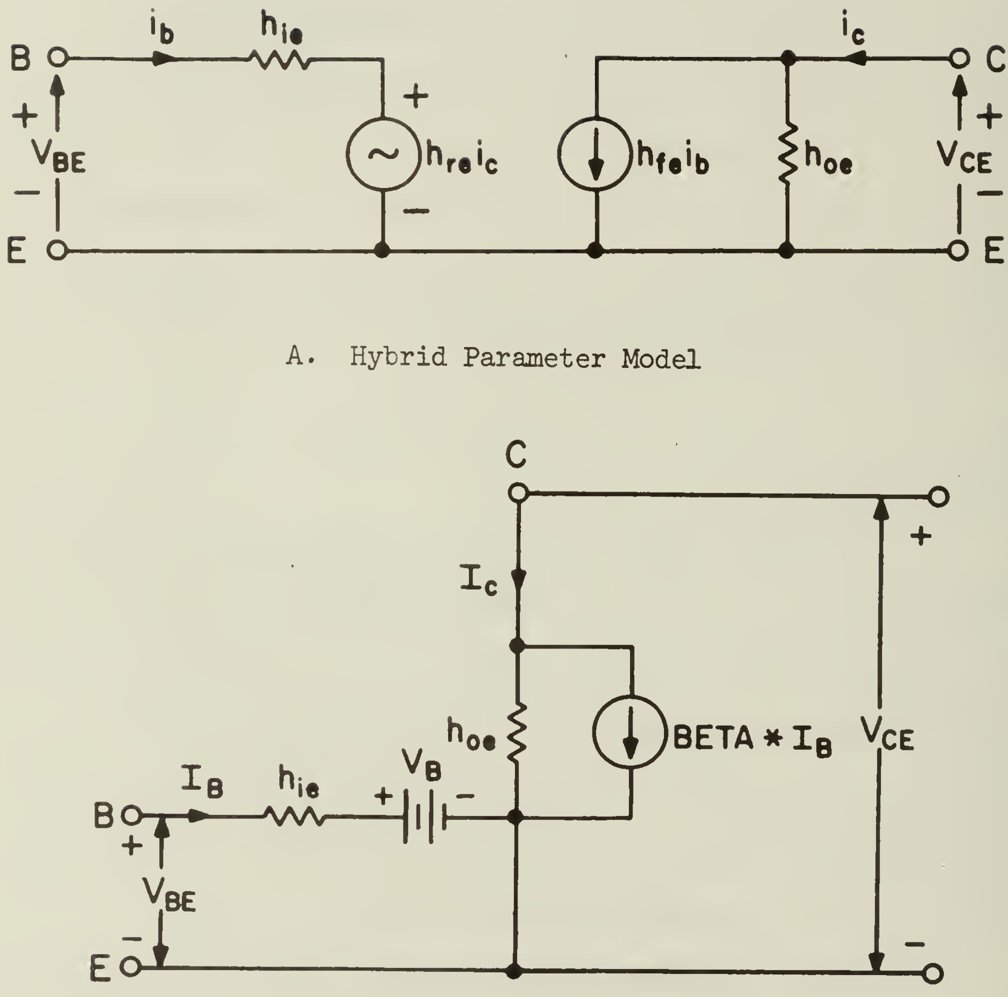
The above intuitively derived nonlinear model is easily extended to include inverse operation by adding an $\alpha_R i_{D2}$ generator, as suggested by the Ebers Moll Model, plus a shunting ideal resistor R_E to facilitate modeling of the reverse characteristic slope, of physical transistors, as shown in the third quadrant of Figure 6.

The quantities, I_{01}, I_{02}, α_F and R_C, in the above equations are variables and, therefore, need to be evaluated experimentally. Malmberg[23] states that evaluation of these variables requires one to make a substantial number of measurements.

2.5 <u>The Regression Model</u>

The Regression Model combines the simplicity of the Hybrid parameter model with the nonlinear modeling flexibility of the nonlinear model utilizing the ideal junction diode as discussed in the previous section. No longer is one restricted to an incremental variation about the "static" or quiescent operating point as is true with the unmodified Hybrid parameter model of Figure 8. The operating range is extended by equations for BETA (I_C) and V_{BE} (I_C).

The first step in the modification process is to convert the Hybrid parameter model of Figure 6A to the ECAP direct current model of Figure 8. This is accomplished by substituting a battery V_B for the $h_{re} \cdot i_c$ voltage generator and substituting the direct current gain, BETA, for the small signal forward current gain, h_{fe}.



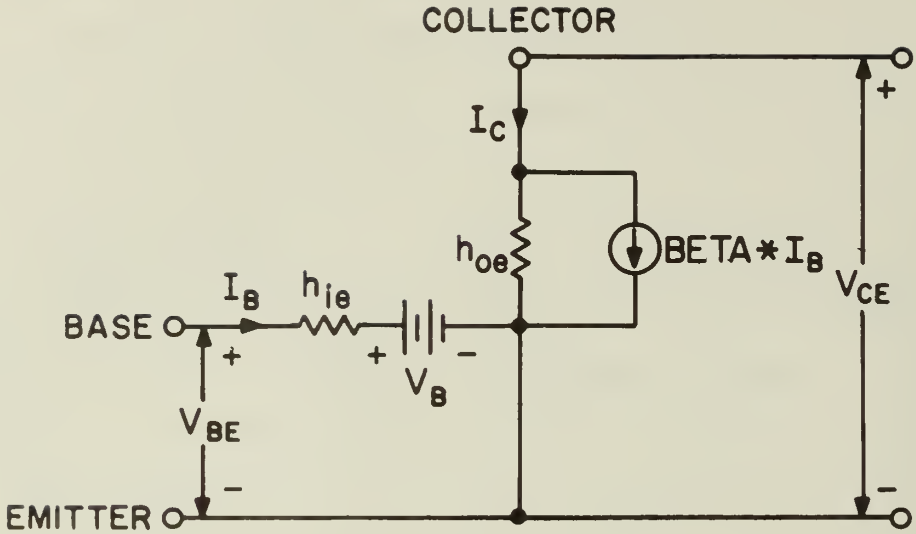
A. Hybrid Parameter Model

B. ECAP Direct Current Model

Figure 8. Comparison of Hybrid and ECAP Transistor Models

The final step involves the use of modeling equations developed by Guth[24] but in normalized form so that the natural logarithm is taken of a dimensionless quantity. These modeling equations control the base to emitter voltage drop V_{BE} and the current gain BETA of the ECAP model. The advantage of this approach is that the circuit analyst need not use the ideal junction diode as a modeling element. Instead of the exponential function and its attendant numerical evaluation problems when used in matrix equations, the user may attain the equivalent[25] circuit simulation range from curves derived from either measured or manufacturer's data points. The data points are fed into the Linear Regression curve fitting program of Appendix C for the purpose of generating in closed form, a mathematical equation which describes the actual variations in V_{BE} and BETA as a function of operating point (collector current, I_C) for each transistor used. In this manner the nonlinear voltage-current relations of any transistor may, in general, be established for any desired operating range below saturation.

Note that the nonlinear Regression Model of Figure 9 is an extension of Guth's model for the purpose of permitting simulation of transistor circuit operation in both the active and cutoff regions as is required, for example, for modeling nonsaturating Emitter Coupled Logic circuitry. The modification consisted of adding the program logic required to switch to a zero valued function of collector current, I_C, whenever the base current, I_B, decreased below a preselected reference level, I_O, or reversed sign from the normal forward direction of positive amplification. Also, the base circuit is modified by setting



A) ECAP Direct Current Model

$$BETA = B_1 + B_2 \left(\ell n \frac{I_C}{I_0}\right) + B_3 \left(\ell n \frac{I_C}{I_0}\right)^2$$

$$V_{BE} = V_1 + V_2 \left(\ell n \frac{I_C}{I_0}\right) + V_3 \left(\ell n \frac{I_C}{I_0}\right)^2$$

B) Regression Equations

Figure 9. Regression Model With Equations

this circuit equal to a branch with $V_{BE} = V_\gamma$, where V_γ is the cutin voltage, in series with a resistance of five megohms, whenever $I_B \leqq 0$.

Thus the solution set of the Regression Model is now restricted to positive values of BETA, I_C, and I_B, by the addition of a conceptual transistor base-emitter diode which behaves in the cutoff mode in a similar manner to the ideal junction diode of the Ideal Diode Model constructed in Section 2.4 above.

The application of this Regression Model requires the user to calculate BETA using the equation:

$$\text{BETA} = \frac{I_{C2} - I_{C1}}{I_{B2} - I_{B1}}$$

where BETA is defined as the spacing[26] between the common emitter output characteristic curves of the Ebers Moll Model operating in the normal active region. There is an implied extension of this definition of BETA required in the use of this model. For example, to extend the Regression Model's validity into the cutoff region, it was necessary to obtain BETA and V_{BE} data at low currents where the output characteristic curves are highly nonlinear. However, the Regression Model specifically allows BETA to be a variable whereas in the Ebers Moll Model BETA is defined as a constant. The use of the above equation is justified for currents which are near zero, since the collector current, I_C, approaches zero at a much greater rate than does the base

current, I_B. In the limit the above equation does not converge to zero as desired, but it is a simple matter to define BETA as being zero at $I_C = 0$ and $I_B = 0$, since this is consistent with all physical observations of BETA's behavior.

The base to emitter forward diode drop, V_{BE}, voltage data is obtained from the common emitter input characteristic curves. Thus, the user supplies as input to the Linear Regression program a table of I_C, BETA, and V_{BE} which contains sufficient data to completely define each transistor in the region to be modeled.

The accuracy with which the Regression Model, in conjunction with the ECAP program, will simulate circuit operation is directly affected by the "goodness of fit" of the regression equations to the experimental data. A plot of typical measured physical variations of V_{BE} and BETA as a function of the collector current, I_C, for the comparison with the Linear Regression prediction equations, is shown in Figures 10 and 11, respectively. The Linear Regression program of Appendix C was used to derive these equations. A convenient program to test the Regression Model's effectiveness is given in Appendix D.

In conclusion, this discussion of models has shown the relationship of the Regression Model to other types of transistor models currently in wide use. It is believed that this model will form the basis of a useful parameter variation simulation program.

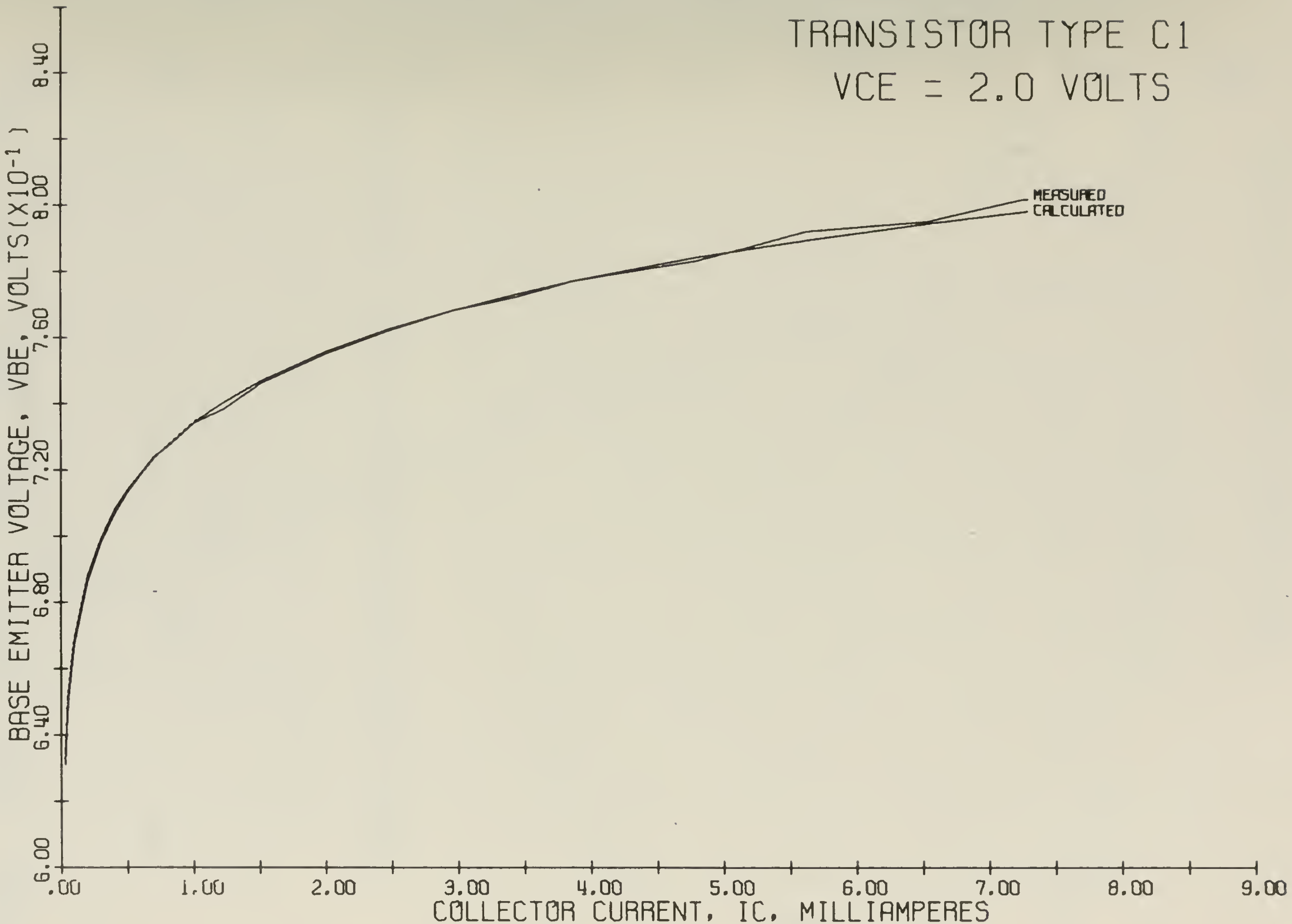


Figure 10. C1 Transistor Base to Emitter Voltage Versus Collector Current

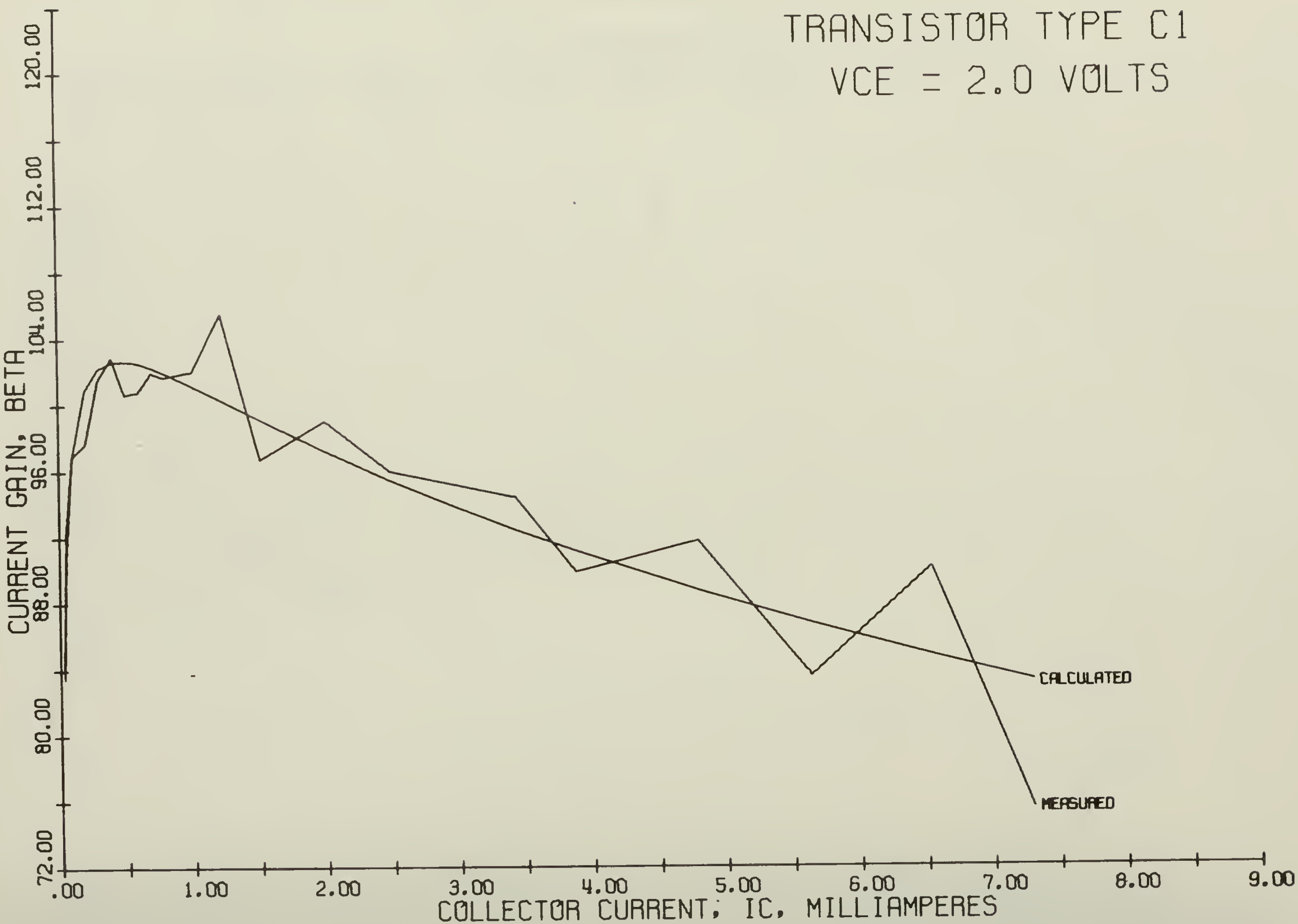


Figure 11. C1 Transistor Current Gain Versus Collector Current

3. COMPARISON OF TWO COMPUTER PROGRAM CIRCUIT ANALYSIS METHODS

3.1 Discussion of a Basic Circuit Analysis Problem

A problem often circumvented in the analysis of transistor circuits is the current convergence problem. This problem arises from the appearance of a dependent generator in the network model and the simultaneous need to satisfy the KCL and KVL equations. Specifically, whenever the circuit external to the circuit model has a direct influence upon the current gain, BETA, of one or more states, then the proper value of BETA must be determined prior to the completion of circuit solution. Two approaches for solving this problem are: the Matrix Partition Method and the Regression Model Method.

Both of the above methods utilize the matrix circuit analysis procedures as discussed by Seshu[27]. For ease of notation interpretation it is suggested that one refer to the matrix analysis[28] of a simple circuit which is presented in Appendix A.

3.2 Solution of the Current Convergence Problem - Matrix Partition Method

The circuit of Figures 12A and 12B is a proportional controlled oven heater driving amplifier used for temperature stabilization of a crystal controlled computer timing oscillator. Using the methods of Appendix A, it is possible to solve for all branch and element currents.

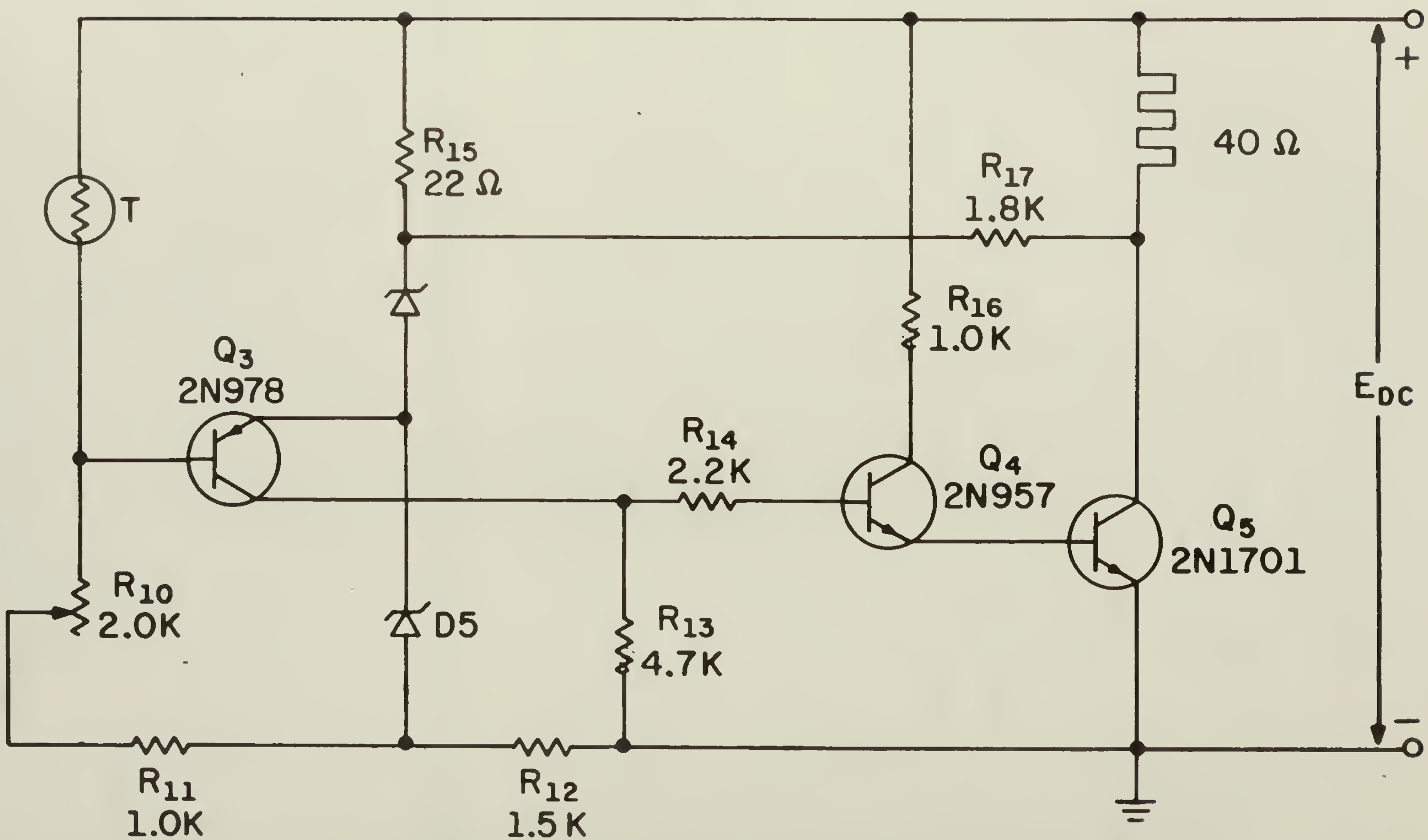


Figure 12A. Oven Heater Driving Amplifier Schematic Diagram

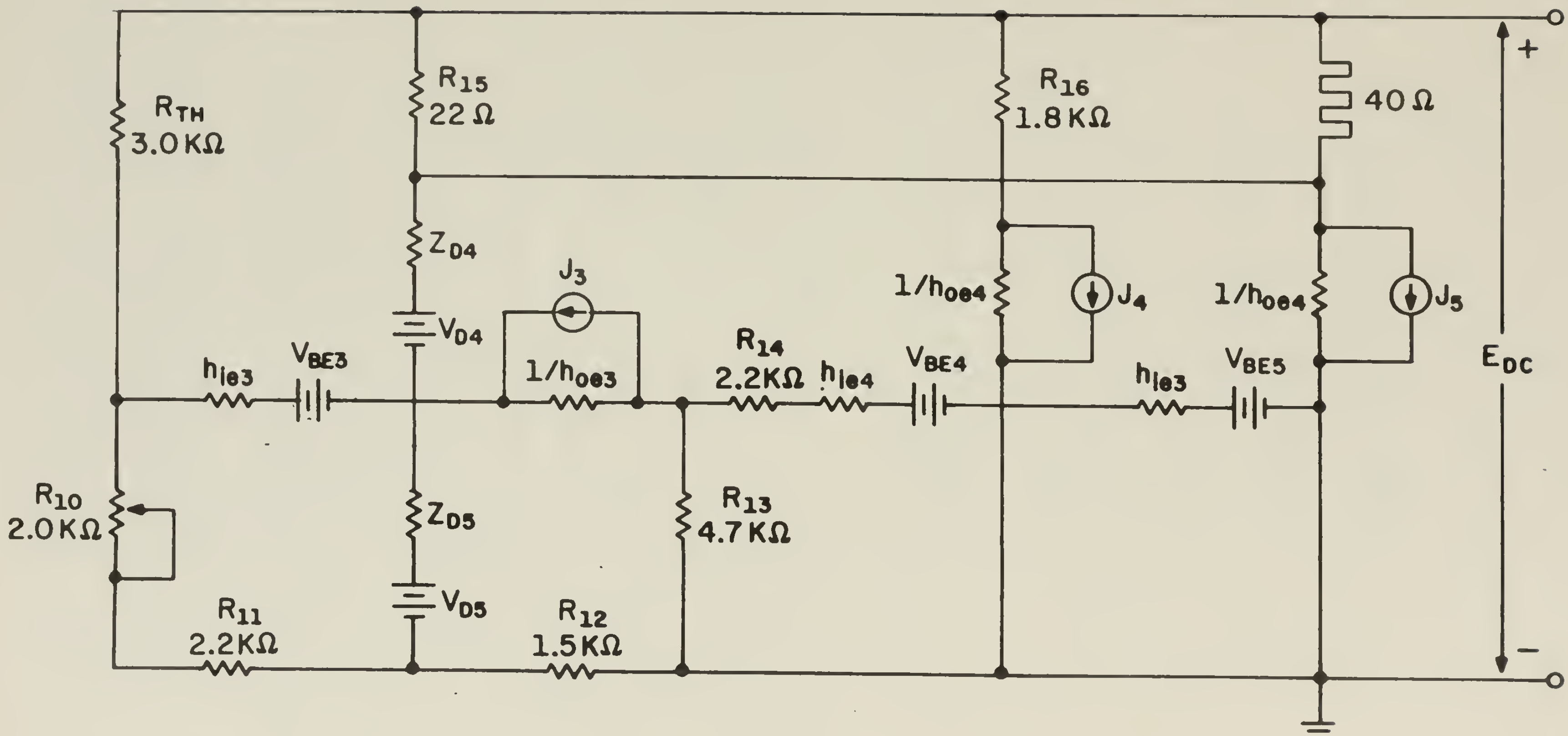


Figure 12B. Oven Heater Driving Amplifier Equivalent Circuit

The matrix equations are:

$$I_B = I_M - JG$$

$$= A^T * I_M - JG$$

$$= A^T * Y_M * E_M - JG$$

$$= [A^T * Y_M * A] * E - JG$$

$$= [A^T * Y_M * A] * [E_B * E_N] - JG$$

$$= [A^T * Y_M * A] * E_B + [A^T * Y_M * A] * E_M - JG$$

$$= [A^T * Y_M * A] * E_B + [A^T * Y_M * A] * Z * JG - JG$$

The dependent collector current generator matrix, JG, and the independent voltage source matrix, E_B, are shown in Figures 13 and 14, respectively. Note that the elements of the JG matrix are factored to show the dependent relationship between the elements of this matrix and the i_{Bj} elements of the I_B matrix for which the solution is being obtained. Therein is the current convergence problem, referred to above, in that certain of the unknown currents appear on both sides of the matrix equations. However, through the use of matrix partitioning, the dependent current values, I_{Bd}, may be obtained and the remaining independent current values, I_{Bi}, calculated. The procedure is:

Let $J = Q * I_n$ where $Q = [q_n]_{k \times 1}$

and q_n = current gain of the n^{th} transistor, $(\text{BETA})_n$.

$$J_D = \begin{bmatrix} 0 \\ h_{FE5}\,(-i_{B5}) \\ 0 \\ h_{FE4}\,(-i_{B4}) \\ 0 \\ h_{FE3}\,(-i_{B3}) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \text{ Amperes}$$

Figure 13. Dependent Current Generator Matrix

$$E_B = \begin{bmatrix} 0 \\ 0 \\ -V_{BE5} \\ 0 \\ -V_{BE4} \\ 0 \\ 0 \\ 0 \\ 0 \\ -V_{D5} \\ 0 \\ -V_{D4} \\ -V_{BE3} \\ 0 \\ 0 \\ 0 \\ 0 \\ E \end{bmatrix} \text{ Volts}$$

Figure 14. Independent Voltage Source Matrix

Thus given the A and Z matrices in Figures 15 and 16 one may solve for the branch current matrix, I_B, where:

$$I_B = [A^T*Y_M*A]*E_B + [A^T*Y_M*A]*Z*Q*I_{Bd} - Q*I_{Bd}.$$

By partitioning the I_B matrix into dependent and independent parts one obtains:

$$I_B = \left[\frac{I_{Bd}}{I_{Bi}}\right] = Y*\left[\frac{\Phi}{E_B}\right] + Y*\left[\frac{Z*Q*I_{Bd}}{\Phi}\right] - \left[\frac{Q*I_{Bd}}{\Phi}\right],$$

where Φ is an appropriately dimensioned null vector in each case. Next the network admittance of Y matrix, where

$$Y = A^T*Y_M*A$$

may be partitioned into four parts:

$$Y_1,\ Y_2,\ Y_3,\ \text{and } Y_4$$

having the respective dimensions of:

$$3 \times 3,\quad 3 \times 15,\quad 15 \times 3,\quad \text{and}\quad 15 \times 15$$

for the 18 branch equivalent circuit of Figure 12-B. The branch current matrix, upon partitioning the Y matrix, becomes:

$$I_B = \left[\frac{I_{Bd}}{I_{Bi}}\right] = \left[\begin{array}{cc} Y_1 & Y_2 \\ \hline Y_3 & Y_4 \end{array}\right] \frac{\Phi}{E_B} + \left[\begin{array}{cc} Y_1 & Y_2 \\ \hline Y_3 & Y_4 \end{array}\right]\left[\frac{Z*Q*I_{Bd}}{\Phi}\right] - \left[\frac{Q*I_{Bd}}{\Phi}\right].$$

$$A = \begin{bmatrix} -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\ 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & -1 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 1 & -1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Figure 15. Coefficient Matrix Element Values

$$= \begin{bmatrix}
R_H & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{1}{h_{oe5}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & h_{ie5} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{1}{h_{oe4}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & h_{ie4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & R_{17} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{h_{oe3}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & R_{13} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & R_{12} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & Z_{D4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & R_{15} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & Z_{D5} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & h_{ie3} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & R_{10}+R_{11} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & R_{TH} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & R_{16} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & R_{14} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & R_S
\end{bmatrix} \text{OHMS}$$

Figure 16. Impedance Matrix Element Values

From this latter equation one may obtain the relations for I_{Bd} and I_{Bi}.

$$I_{Bd} = Y_2*E_B + Y_1 *[Z*Q*I_{Bd}] - Q*I_{Bd}$$

or $$I_{Bd} = [U - Y_1*Z*Q + Q]^{-1} *[Y_2*E_B]$$

and $$I_{Bi} = Y_4*E_B + Y_3*Z*Q*I_{Bd};$$

where U is the identity matrix having ones at each position along the major diagonal. Two points should be noted. First, it is necessary to develop a permutation matrix for transforming the original matrices into the desired partitive form and, second, the user must supply a value for current gain, BETA, of each transistor. From the numerical analysis viewpoint, there is no formal algorithm for deriving the permutation matrix; however, through the use of cyclical iterative analysis methods[29] the exact value of BETA for each transistor may be woven into the solution provided one has an equation relating the current gain, BETA, to the base of collector current of each transistor.

3.3 Solution of the Current Convergence Problem - Regression Model Method

In contrast to the solution of the previous section for the network branch currents, the Regression Model Method utilizes an existing computer circuit analysis program, ECAP[30], which first solves for the network node voltages, and second, solves for the resulting element voltages and currents.

Mathematically speaking both methods utilize the same matrix formulation in that the A matrix approach is used for converting the branch currents into a statement of the total currents entering each node as required by Kirchoff's Current Law (KCL).

The basic cyclical iterative analysis procedure developed by Guth[31] in conjunction with the IBM 1620 version of ECAP was not changed upon coupling the modified iterative subroutine of Appendix B into the new IBM 360 version of ECAP[32]. As an aid to understanding mathematical processes used, the matrix equations for the test circuit of Figure 17 will be developed to show where Guth's cyclical iterative technique enters the circuit analysis computational process.

The matrix method of formulating circuit analysis uses a fundamental building block called the branch element. The ECAP[33] standard branch element is sketched in Figure 18. Note that this branch contains provisions for all possible driving functions as well as the corresponding response functions. The branch terminal voltage, V_B, is given by the matrix equation:

$$V_B = V_R - E$$

where V_R and E are the resistive element voltage and the voltage source column matrices, respectively.

The branch currents, I_B, flowing out of the branches, with the positive direction defined as being from the m^{th} node to the n^{th} node are given by the matrix equation:

$$I_B = I_R + J$$

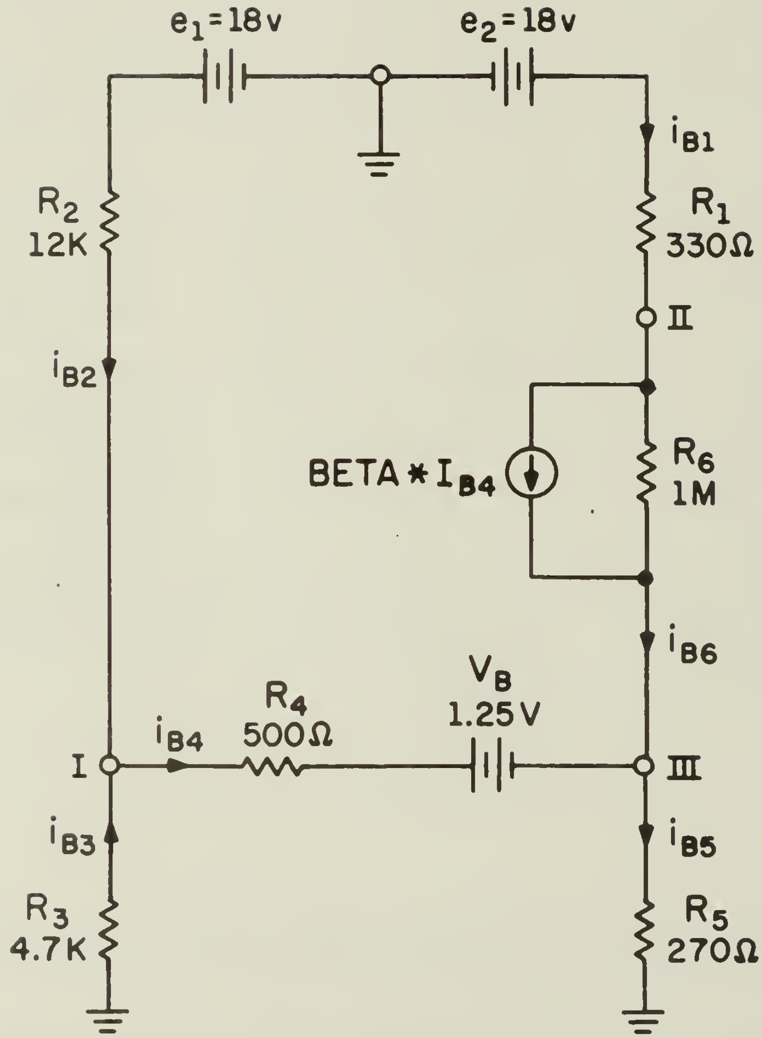


Figure 17. NLAP Common Emitter Test Circuit

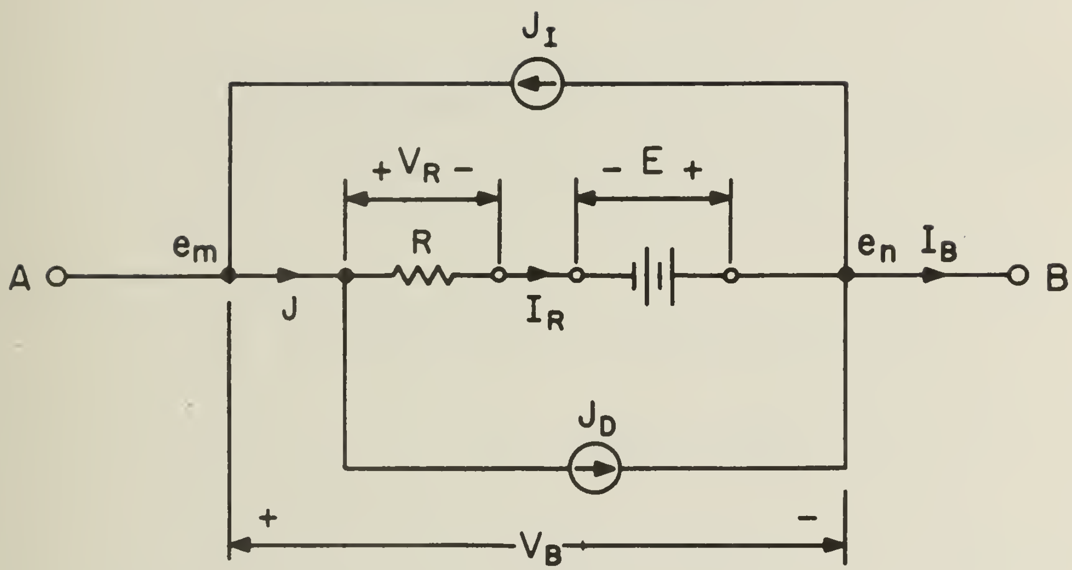


Figure 18. ECAP Standard Branch

where I_R is the current through the branch admittance element, Y_{ij}, of the circuit admittance matrix, Y, and J is the sum of the quantities J_D and J_I, which are the branch dependent and independent current generator matrices, respectively.

The network driving voltages, E, may be replaced, using Norton's Theorem, by the corresponding driving currents. Thus if E is the voltage source column matrix, the transformed source current is:

$$J' = Y * E$$

and the total driving current in the network becomes:

$$J_T = J_I + J_D + Y*E.$$

Kirchoff's Current Law in matrix form is:

$$I_N = A*I_B = 0$$

where now $$I_B = I_R + J_T,$$

and thus, by expansion:

$$I_N = A * I_B = [A * I_R] + [A * J_T].$$

Consequently: $$A * Y * V_R = [-A] * J_T.$$

From Kirchoff's Voltage Law in matrix form:

$$V_R = A^T * V_N.$$

Combining the two previous equations:

$$[A * Y * A^T] * V_N = [-A] * J_T;$$

and solving for the node voltage matrix, V_N:

$$V_N = [A * Y * A^T]^{-1} * [-A] * [J_T]$$

$$= [A * Y * A^T]^{-1} * [-A] * [J_I + J_D + Y * E].$$

Following through now with the illustrative example, we begin by writing the KCL equations for the circuit of Figure 17.

$$\begin{array}{cccccccc} 0 & -i_2 & -i_3 & -i_4 & 0 & 0 & = & 0 \\ -i_1 & 0 & 0 & 0 & 0 & +i_6 & = & 0 \\ 0 & 0 & 0 & -i_4 & +i_5 & -i_6 & = & 0 \end{array}$$

These equations become in matrix notation $I_N = A * I_B$, or

$$I_N = \begin{bmatrix} I_{n1} \\ I_{n2} \\ I_{n3} \end{bmatrix} = \begin{bmatrix} 0 & -1 & -1 & +1 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & +1 \\ 0 & 0 & 0 & -1 & +1 & -1 \end{bmatrix} * \begin{bmatrix} i_{B1} \\ i_{B2} \\ i_{B3} \\ i_{B4} \\ i_{B5} \\ i_{B6} \end{bmatrix}$$

The voltage and current source matrices are, respectively:

$$E = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \end{bmatrix} = \begin{bmatrix} 18.0 \\ 18.0 \\ 0 \\ V_B \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 18.0 \\ 18.0 \\ 0 \\ -1.25 \\ 0 \\ 0 \end{bmatrix}$$

and:

$$J_T = J_I + J_D + Y*E$$

$$J_T = \Phi + \begin{bmatrix} j_{D1} \\ j_{D2} \\ j_{D3} \\ j_{D4} \\ j_{D5} \\ j_{D6} \end{bmatrix} + [Y*E]$$

$$J_T = \Phi + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \text{BETA} * i_{B4} \end{bmatrix} + [Y] * \begin{bmatrix} 18.0 \\ 18.0 \\ 0 \\ -1.25 \\ 0 \\ 0 \end{bmatrix}$$

Note that the voltage source matrix, E, clearly shows the role of V_{BE} and that the dependent current generator matrix, J_D, clearly shows the role of the transistor's current gain, BETA. These are the two quantities which are iteratively changed by the Regression Model. However, it is very difficult to solve the matrix equation,

$$V_N = [A*Y*A^T]^{-1} * [-A] * [J_I + J_D + Y*E],$$

when the dependent current BETA*i_{B4} is not known. For this reason, it is necessary to modify the above equation slightly.

The requisite modification is obtained by writing the node equations for the circuit of Figure 17. Upon substituting the equivalent expression for i_{B4} in terms of V_1, V_3, V_B, and R_4 and collecting terms one obtains the matrix relation:

$$Y_N * V_N = J_T$$

where Y_N is the nodel conductance matrix[34] of Figure 19 and satisfies the equation:

$$Y_N = [A*Y*A^T]$$

which was discussed above. The corresponding nodal equivalent current vector is given in Figure 20. This current vector satisfies the relation:

$$J_T = J_I + J_D + Y*E$$

which was derived above.

$$Y_N = \begin{bmatrix} (1/4700\ \Omega + 1/500\ \Omega + 1/12000\ \Omega) & 0.0 & -(1/500\ \Omega) \\ (\mathrm{BETA}/500\ \Omega) & 1/330\ \Omega + 1/10^7\ \Omega & -(\mathrm{BETA}/500\ \Omega + 1/10^7\ \Omega) \\ -(\mathrm{BETA}/500\ \Omega + 1/500\ \Omega) & 0.0 & (\mathrm{BETA}/500\ \Omega + 1/500\ \Omega + 1/270\ \Omega + 1/10^7\ \Omega) \end{bmatrix}$$

Figure 19. Nodal Conductance Matrix with BETA Explicitly Shown

$$J_T = \begin{bmatrix} 18/12000\ \Omega & + & V_{BE}/500\ \Omega \\ 18/330\ \Omega & + & V_{BE}/500\ \Omega \\ -V_{BE}/500 & - & V_{BE} * BETA/500\ \Omega \end{bmatrix}$$

Figure 20. Nodal Equivalent Current Vector With BETA and V_{BE} Explicitly Shown

3.4 Cutoff Conditions and the Regression Model

The conditions which must be satisfied for each transistor in a network, which is in a nonconducting or cutoff state, are BETA equal to zero and a base to emitter voltage, V_{BE}, less than or equal to the cutin voltage[35], V_γ. The condition of BETA = 0 implies that the collector current, I_C, is zero, provided the collector leakage amount I_{CO}, is negligible, since at low values of I_C:

$$\text{BETA} \cong \frac{\text{Total } I_C}{\text{Total } I_B}$$

This further means that, when there is no significant base current flowing, V_{BE} is determined by the circuits external to each transistor.

From Figure 9, the Regression Model equations are:

$$\text{BETA} = B_1 + B_2 \left(\ell n \frac{I_C}{I_0}\right) + B_3 \left(\ell n \frac{I_C}{I_0}\right)^2$$

$$V_{BE} = V_1 + V_2 \left(\ell n \frac{I_C}{I_0}\right) + V_3 \left(\ell n \frac{I_C}{I_0}\right)^2$$

Since the ECAP model is a strictly linear network, the currents in the base, emitter and collector circuit branches may flow in either the conventional positive direction for an NPN transistor or in the opposite direction.

A transistor's current gain, BETA, may become zero in one of three ways. First, BETA may be made zero by using a logical operator statement, such as the FORTRAN IV code:

```
IF (I_C .LE. CUTOFF) BETA = 0.0
```

placed in Subroutine ITRATE after the BETA Regression Model equation has been evaluated. Second, it may become zero by carefully selecting the BETA regression equation which predicts a zero value for BETA at a suitably low value for the collector current, I_C. Third, a combination of the previous two methods may be used. The first method tends to generate discontinuities which make convergence difficult whenever CUTOFF is an arbitrarily selected current level. The second method gives a smooth curve down to very large negative values of BETA, as I_C becomes increasingly smaller. An example of this event is shown in Table 1. Note that in the BETA equation of Table 1, CUTOFF corresponds to a value of I_C somewhere between 1.0 and 12.0 microamperes and I_0 corresponds to 1.0 microampere. The third method involves a combination of the simplicity of the previous two, with CUTOFF chosen from the BETA equation and with the "IF statement" inhibiting negative values of BETA.

A transistor's base to emitter voltage, V_{BE}, is a function of I_C throughout the active region. The quality of fit for a typical Regression Model V_{BE} equation is shown in Table 2. Note that with the same value of CUTOFF used for calculating V_{BE}, one obtains a number close to V_γ. However, as the drive to any transistor falls below $V_{BE} = V_\gamma$, the voltage across its emitter to base terminals becomes a function of the external circuitry surrounding it. Thus at CUTOFF and below, a logical switch needs to be incorporated into Subroutine ITRATE; so that the V_{BE} may be a function of some external circuit current, I_α. A better external circuit dependent variable would be

BETA = -144.30034 + 76.79433 * LN (IC) + -5.76604 * (LN(IC)) **2

VCE VOLTS	IC (MICROAMPS)	MEASURED BETA (--)	CALCULATED BETA (--)	BETA MEAS-CALC (--)
3.00	1.00000	0.0	-144.30034	144.30034
3.00	12.00000	10.00000	10.92245	-0.92245
3.00	24.70000	42.59999	42.66835	-0.06836
3.00	38.29999	90.66699	59.02272	31.64427
3.00	62.09999	95.20000	74.47285	20.72714
3.00	112.29999	100.39999	89.73676	10.66324
3.00	213.50000	101.20000	101.71530	-0.51530
3.00	320.79980	107.29999	106.84273	0.45726
3.00	428.29980	107.50000	109.32199	-1.82199
3.00	535.29980	107.00000	110.57671	-3.57671
3.00	642.39990	107.09999	111.17651	-4.07652
3.00	748.79980	106.39999	111.38388	-4.98389
3.00	854.79980	106.00000	111.34521	-5.34521
3.00	1079.00000	112.09999	110.78613	1.31386
3.00	1287.99878	104.50000	109.94238	-5.44238
3.00	1589.00000	100.33400	108.47314	-8.13914
3.00	2113.00000	104.79999	105.66577	-0.86578
3.00	2620.00000	101.39999	102.92725	-1.52725
3.00	3117.99878	99.59999	100.32104	-0.72105
3.00	3608.00000	98.00000	97.86548	0.13452
3.00	4094.99878	97.39999	95.53638	1.86362
3.00	5054.99219	96.00000	91.25317	4.74683
3.00	5929.99219	87.50000	87.66528	-0.16528
3.00	6864.99219	93.50000	84.11670	9.38330
3.00	7666.99219	80.20000	81.27490	-1.07491

Table 1. Analysis of Regression Equation for BETA, C_1 $V_{CE} = 3.0\ V_{DC}$

VBE = 0.49696 + 0.03842 * LN (IC) " -0.00058 * (LN(IC)) **2

VCE VOLTS	IC (MICROAMPS)	MEASURED VBE (--)	CALCULATED VBE (--)	VBE MEAS-CALC (--)
3.00	1.00000	0.60300	0.49696	0.10604
3.00	12.00000	0.60300	0.58885	0.01415
3.00	24.70000	0.60300	0.61420	-0.01120
3.00	38.29999	0.63200	0.62931	0.00269
3.00	62.09999	0.65200	0.64570	0.00630
3.00	112.29999	0.67200	0.66542	0.00658
3.00	213.50000	0.69100	0.68635	0.00465
3.00	320.79980	0.70300	0.69936	0.00364
3.00	428.29980	0.71000	0.70848	0.00152
3.00	535.29980	0.71600	0.71545	0.00055
3.00	642.39990	0.72000	0.72111	-0.00111
3.00	748.79980	0.72400	0.72584	-0.00184
3.00	854.79980	0.72800	0.72990	-0.00190
3.00	1079.00000	0.73400	0.73699	-0.00299
3.00	1287.99878	0.73700	0.74234	-0.00534
3.00	1589.00000	0.74600	0.74864	-0.00264
3.00	2113.00000	0.75500	0.75710	-0.00210
3.00	2620.00000	0.76200	0.76343	-0.00143
3.00	3117.99878	0.76700	0.76851	-0.00151
3.00	3608.00000	0.77200	0.77274	-0.00074
3.00	4094.99878	0.77600	0.77639	-0.00039
3.00	5054.99219	0.78200	0.78243	-0.00043
3.00	5929.99219	0.79000	0.78697	0.00303
3.00	6864.99219	0.79400	0.79110	0.00290
3.00	7666.99219	0.80000	0.79421	0.00579

Table 2. Analyis of Regression Equation for V_{BE}, C_1 $V_{CE} = 3.0\ V_{DC}$

a voltage, but this leads to the matrix solution problem similar to that discussed in Section 3.2, where now node voltage is the independent variable. Note that both Table 1 and Table 2 were calculated using the Regression Model Evaluation Program listed in Appendix D.

3.5 Extension of the Regression Model

Two approaches requiring only minor Regression Model modification were explored. First, V_{BE} was set equal to the cutin voltage with the result that satisfactory predictions of network behavior can be made for operating conditions close to cutoff. Obvious inaccuracies are introduced as each transistor's operating point passed well into the cutoff state. The second approach consisted of adding a large impedance, ZOFF, in series with the base circuit of the transistor for the purpose of simulating ideal diode behavior. The ZOFF impedance of 5,555,555.0 was switched into the circuit by a logical operation statement in Subroutine ITRATE whenever the I_C current became less than I_O. This latter approach was tested using the circuit of Figure 1-B with the result that current convergence was not obtainable.

A check was made of the effects of setting BETA equal to zero and V_{BE} equal to V_γ, by examining the matrices used in the matrix equation:

$$Y_N * V_N = J_T.$$

The solution to this set of simultaneous equations will exist provided the Y_N matrix has an inverse. By inspection of Figure 19, it is apparent that this will always be the case provided the admittances are real and positive; since BETA appears only in nonmajor diagonal terms. The possible range of variations in V_{BE} also will not affect the existance of Y_N inverse since V_{BE} only appears in the J_T matrix of Figure 20. This means a solution for V_N will exist even if J_T is a zero vector.

3.6 Conclusion

Summarizing briefly, the computer program NLAP solves for the linear current in each of the branches. The resulting collector current is then used as the dependent variable in the Regression Model equations of Figure 9 by Subroutine ITRATE. The iterative process is started in accordance with Guth's flow chart[36] of Figure 21 until convergence. Then the proper values of BETA and V_{BE} are obtained as a function of the collector current in the active region. The calculations made by the program may be checked both before iteration starts and after convergence is obtained by using the PRINT MI input code described in 4.3. A set of matrices obtained in this are shown in Figures 22 and 23, respectively.

One interesting example of the accuracy of NLAP is a DC option analysis of the linear test circuit of Figure 17. The NLAP computed results are compared with manually calculated results, for the nodal conductance matrix, in Figure 24, and for the equivalent current vector matrix in Figure 25. The results compare closely out to the sixth decimal place.

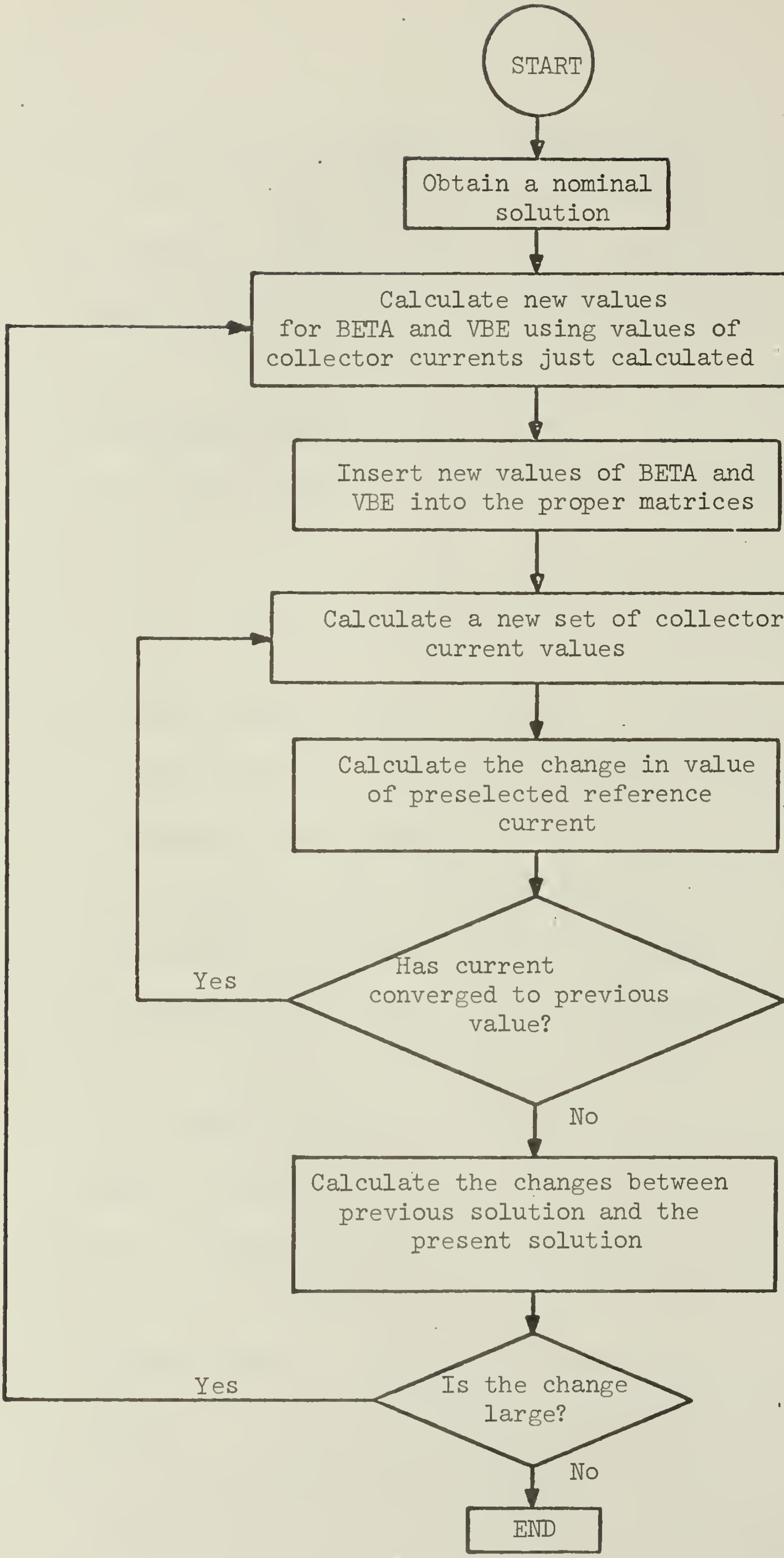


Figure 21. Flow Chart of Iterative Regression Model Solution Method.

NODAL CONDUCTANCE MATRIX

ROW	COLS				
1	1 - 4	0.238679200-01	-0.49999990D-02	0.0	0.0
1	5 - 5	0.0			
2	1 - 4	-0.50499988D 00	0.10050732D 03	-0.13333332D-03	-0.21758056D-02
2	5 - 5	-0.10000002D 03			
3	1 - 4	0.49999988D 00	-0.50013321D 00	0.10000015D 03	0.0
3	5 - 5	0.0			
4	1 - 4	0.0	-0.21758056D-02	0.0	0.10000219D 03
4	5 - 5	0.0			
5	1 - 4	0.0	-0.10000002D 03	0.0	0.0
5	5 - 5	0.10000003D 03			

EQUIVALENT CURRENT VECTOR

NODE NO.	CURRENT
1	0.22717916D-01
2	-0.38884987D 00
3	0.11538497D 03
4	-0.29000000D 03
5	0.0

NODAL IMPEDANCE MATRIX

ROW	COLS				
1	1 - 4	0.52935884D 02	0.52172248D 00	0.69562887D-06	0.11351418D-04
1	5 - 5	0.52172243D 00			
2	1 - 4	0.52693899D 02	0.24904866D 01	0.33206435D-05	0.54186960D-04
2	5 - 5	0.24904864D 01			
3	1 - 4	-0.11396619D-02	0.98471243D-02	0.99999983D-02	0.21424959D-06
3	5 - 5	0.98471233D-02			
4	1 - 4	0.11464917D-02	0.54186960D-04	0.72249166D-10	0.99997821D-02
4	5 - 5	0.54186955D-04			
5	1 - 4	0.52693894D 02	0.24904864D 01	0.33206432D-05	0.54186955D-04
5	5 - 5	0.25004861D 01			

Figure 22. DC Solution Matrix Obtained Prior to Iteration

NODAL CONDUCTANCE MATRIX

ROW	COLS				
1	1 - 4	0.23867920D-01	-0.49999990D-02	0.0	0.0
1	5 - 5	0.0			
2	1 - 4	-0.49911976D 00	0.10050144D 03	-0.13333332D-03	-0.21758056D-02
2	5 - 5	-0.10000002D 03			
3	1 - 4	0.49411976D 00	-0.49425310D 00	0.10000015D 03	0.0
3	5 - 5	0.0			
4	1 - 4	0.0	-0.21758056D-02	0.0	0.10000219D 03
4	5 - 5	0.0			
5	1 - 4	0.0	-0.10000002D 03	0.0	0.0
5	5 - 5	0.10000003D 03			

EQUIVALENT CURRENT VECTOR

NODE NO.	CURRENT
1	0.22654877D-01
2	-0.37802932D 00
3	0.11537421D 03
4	-0.29000000D 03
5	0.0

NODAL IMPEDANCE MATRIX

ROW	COLS				
1	1 - 4	0.52935133D 02	0.52783296D 00	0.70377616D-06	0.11484368D-04
1	5 - 5	0.52783291D 00			
2	1 - 4	0.52690314D 02	0.25196555D 01	0.33595354D-05	0.54821605D-04
2	5 - 5	0.25196553D 01			
3	1 - 4	-0.11394418D-02	0.98453338D-02	0.99999983D-02	0.21421063D-06
3	5 - 5	0.98453328D-02			
4	1 - 4	0.11464137D-02	0.54821605D-04	0.73095358D-10	0.99997821D-02
4	5 - 5	0.54821600D-04			
5	1 - 4	0.52690309D 02	0.25196553D 01	0.33595350D-05	0.54821600D-04
5	5 - 5	0.25296550D 01			

Figure 23. NL Solution Matrix Obtained After Completion of Iteration

$$Y_N = \begin{bmatrix} +0.00229606 & +0.00000000 & -0.00200000 \\ +0.10000000 & +0.00303040 & -0.10000000 \\ -0.10200000 & +0.00000000 & +0.10570380 \end{bmatrix}$$

a) Manual Calculations

$$Y_N = \begin{bmatrix} +0.22960991\text{D-}02 & 0.00000000\text{D+}00 & -0.19999999\text{D-}02 \\ +0.99999964\text{D-}01 & 0.30304029\text{D-}02 & -0.10000000\text{D+}00 \\ -0.10199996\text{D+}00 & -0.99999966\text{D-}07 & +0.10570377\text{D+}00 \end{bmatrix}$$

b) NLAP Calculations

Figure 24. Nodal Conductance Matrix Calculations for BETA = 50

$$J_T = \begin{bmatrix} +0.00400000 \\ +0.17954545 \\ -0.12750000 \end{bmatrix}$$

a) Manual Calculations

$$J_T = \begin{bmatrix} +0.39999976\text{D-02} \\ +0.17954527\text{D+00} \\ -0.12749982\text{D+00} \end{bmatrix}$$

b) NLAP Calculations

Figure 25. Nodal Current Vector Calculations for BETA = 50 and V_{BE} = -1.25 Volts

A second example of the accuracy of NLAP is a NL option analysis, the output of which is compared with laboratory measured performance in the graph of Figure 26 for the emitter follower of Figure 27.

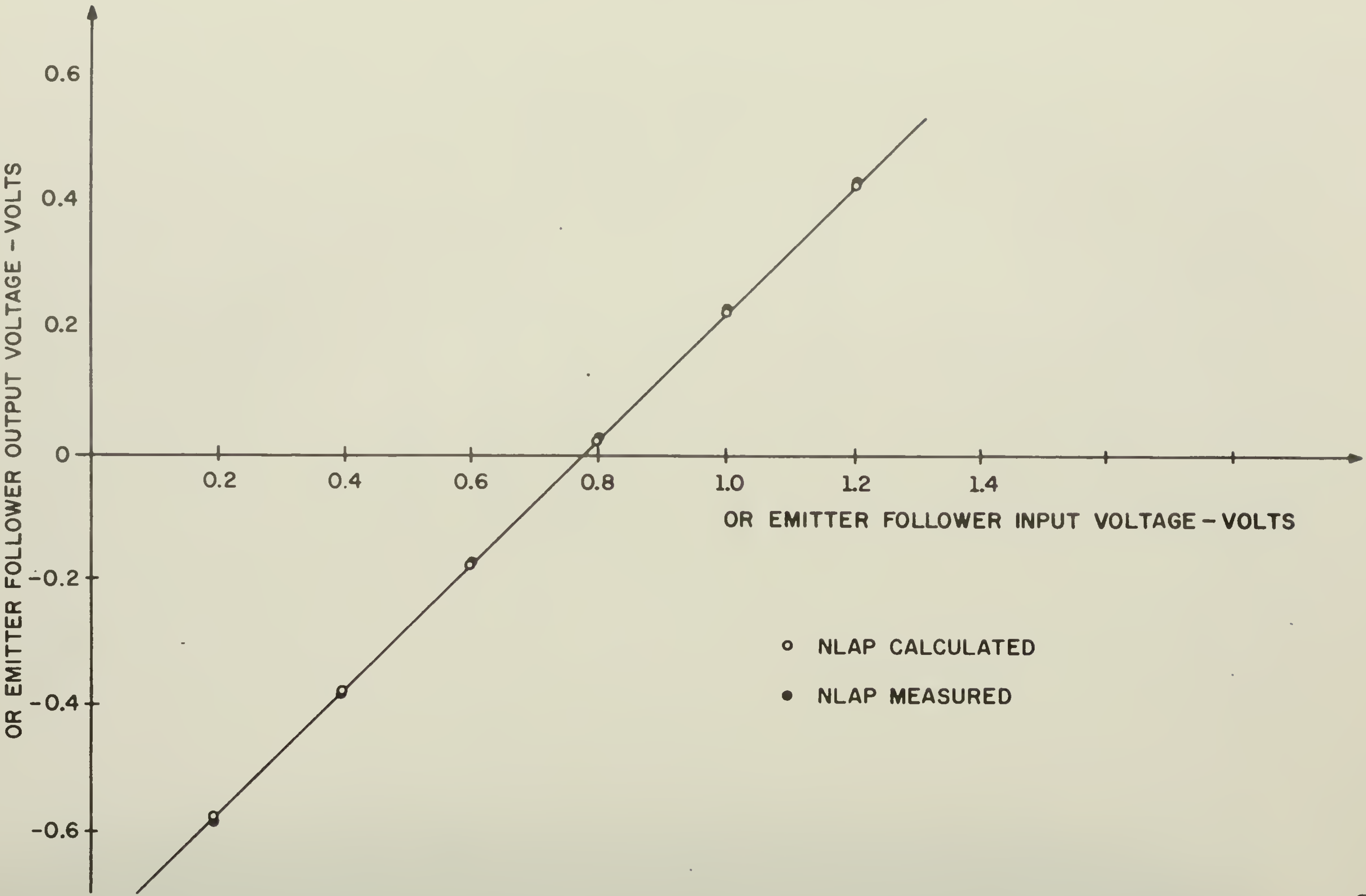


Figure 26. OR Emitter Follower Voltage Offset Characteristic

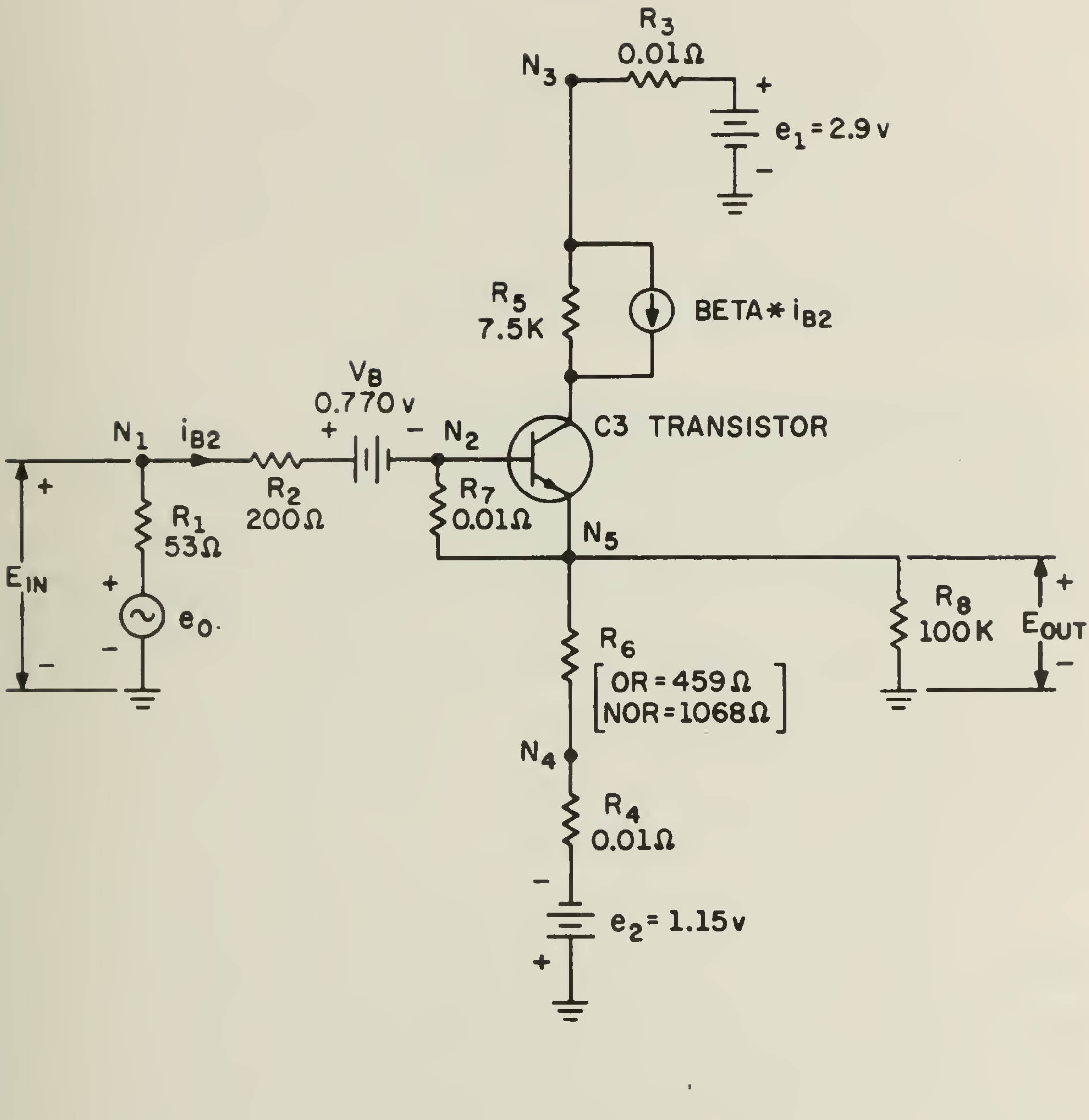


Figure 27. OR Emitter Follower Schematic Diagram

4. APPLICATION OF NLAP TO LOGIC CIRCUIT ANALYSIS

4.1 Introduction

NLAP is intended to be used for analyzing nonsaturating logic circuitry such as emitter coupled logic blocks. The capability now exists within the program for linear region parameter variation analysis. Should the user desire to perform special calculations, a subroutine may be coded and called just prior to the return of control to the main program by Subroutine ITRATE. The NLAP calling procedure and its associated system procedure is given in Figure 28.

4.2 Preparation of NLAP Input Data Deck

The input language to NLAP is similar to ECAP's[36] and is aligned closely with the language of the electronic engineer. Each of the circuit elements use a simple descriptive letter coding and the topology of the circuit interconnections is identified by a sequence of branch and node numbers. The polarity of positive current flow is inserted by ordering the nodal connection using identifying numbers in a from node m to node n arrangement.

The general branch, or B_j, statement consists of a branch number, j; the from and to node numbers, m and n, respectively; the branch resistance, R, in ohms; the independent voltage source E, in volts; and the independent current source I, in amperes. In other words, the general branch statement is:

$$B_j = N(m,n),\ R = \gamma,\ E = e,\ I = i$$

```
/*FORMAT PR,DDNAME=FT10F001,COPIES=1

//JOBLIB  DD  DSNAME=SYS1.CKTLIB,DISP=(OLD,PASS)
//  EXEC  NLAP
//NLAP    EXEC   PGM=NLAP
//               *
//               *    EXECUTE NLAP   --   JOBLIB CARD REQUIRED
//               *    //JOBLIB DD DSNAME=SYS1.CKTLIB,DISP=(OLD,PASS)
//               *
//FT05F001 DD    DDNAME=SYSIN
//FT06F001 DD    SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=798)
//FT07F001 DD    UNIT=(CTC,,DEFER),DCB=(RECFM=F,LRECL=80,BLKSIZE=80)
//FT08F001 DD    UNIT=(CTC,,DEFER),DCB=(RECFM=F,LRECL,80=BLKSIZE=80)
//FT09F001 DD    UNIT=(CTC,,DEFER)=DCB=(RECFM=F,LRECL=80,BLKSIZE=80)
//FT10F001 DD    UNIT=(CTC,,DEFER),DCB=(RECFM=F,LRECL=80,BLKSIZE=80)
//FT11F001 DD    UNIT=(CTC,,DEFER),DCB=(RECFM=F,LRECL=80,BLKSIZE=80)
//FT12F001 DD    UNIT=(CTC,,DEFER),DCB=(RECFM=F,LRECL=80,BLKSIZE=80)
//NLAP.SYSIN  DD  *
                  * NLAP DATA DECK GOES HERE
 /*
```

Figure 28. NLAP Calling Sequence

where j, m, n, γ, e and i are place holders for specific numeric values which vary from branch to branch. An example of the specific coding for the circuit of Figure 27 is listed in Figure 29.

The dependent current sources are denoted by a T_k statement containing the controlling and the controlled branch identification numbers, wotether with the associated current gain value, BETA.

The regression equation coefficients for the BETA and V_{BE} equations are entered in T_k and B_j statements, respectively, which are of the form:

$$T_k \qquad \text{BETA (i)} = B_1, B_2, B_3$$

$$B_j \qquad \text{E(i)} \qquad = V_1, V_2, V_3$$

where T_k and B_j start in card column one with the BETA (β) and the E(β) dependent variable statements starting column seven.

The cyclical iteration process is controlled by two successive statements

$$\text{CURRENT} = \gamma$$

and

$$\text{TOLERANCE} = \delta$$

which imply that the iteration process is to terminate whenever the newest solution for the current in branch γ changes by less than the prespecified amount, δ, from the previous iterations calculated value.

```
ID21   OR   EMITTER FOLLOWER, RE = 459.6 OHMS   -DCAP
       DC
B1     N(0,1),R=53          ,E=1.0
B2     N(1,2),R=200.0,E=-0.770
B3     N(0,3),R=0.01,E=+1.15
B4     N(4,0),R=0.01,E=+2.90
B5     N(3,2),R=7500
B6     N(5,4),R=459.6
B7     N(2,5),R=0.01
B8     N(5.0,R=100000
T1     B(2,5),BETA=100
       PRINT NV,BV,BA,BP,MI
       EX
       END
```

A. DC Circuit Analysis Data Deck

```
ID21   OR   EMITTER FOLLOWER, RE = 459.6 OHMS   -NLAP DATA
       NL
B1     N(0,1),R=53          ,E=+1.0
B2     N(1,2),R=200.0,E=-0.770
B3     N(0,3),R=0.01,E=+1.15
B4     N(4,0),R=0.01,E=+2.90
B5     N(3,2),R=7500
B6     N(2,4),R=459.6
B7     N(2,5),R=0.01
B8     N(5,0),R=100000
T1     B(2,5),BETA=100
       PRINT  NV,MI
       EX
T1     BETA(5)=-554.67056,+172.42553,-11.1718
B2     E(5)=-0.54023,-0.02429,-0.00021
       CURRENT = 6
       TOLERANCE = 0.01
       EX
       RETURN
       END
```

B. NL Circuit Analysis Data Deck

Figure 29. Typical NLAP Input Data

The amount of tolerable deviation from current convergence is stated in a decimal, not percent, format. One example is:

TOLERANCE = 0.01

which means current convergence to within one percent.

4.3 Solution Control Codes

Two types of analyses are possible using NLAP, linear direct current and iterative nonlinear direct current. The solution control cards are DC and NL, respectively.

A DC problem deck must contain only those cards up to the first EX or EXECUTE card as shown in Figure 29-A. The full set of ECAP/360 DC solution options[37] SE, WO, ST, and MI are available to the DC user. These options[38] obtain for the user: element sensitivities, worst case tolerance analysis, standard deviations of the node voltages assuming component values which are statistically independent, random, and normally distributed about their mean values, and a set of solution matrices, respectively.

An NL problem deck must contain the same network description cards as the equivalent DC problem deck, plus the regression equation cards for BETA and V_{BE} and the CURRENT, TOLERANCE, EXECUTE, RETURN and END cards as shown in Figure 29-B. These last three cards are self descriptive and the user should observe that the END card is used to signal the end of the input data so that control may be passed to the system monitor.

It should be noted that the only case where the PR or print card is essential is when the user desires to print out the nodal

conductance matrix, the equivalent current vector, and the nodal impedance matrix. To obtain these matrices the required code is:

```
PRINT NV, MI
```

This coding was used to obtain the matrices shown in Figures 22 and 23.

4.4 Solution Output and Identification

When the user desires to perform a parameter variation analysis, careful attention should be given to the preparation of the title card which is placed at the beginning of each problem deck for solution identification purposes.

The letters ID are placed in card columns one and two for the purpose of signal by the subroutine ITRATE to search for subsequent code. Card column three is reserved for a problem identification number from one to nine which is read and written by the program in A1 format[39]. Card column 5 is reserved for a flag which resets the solution number to one after a series of related solutions have been run for a given problem and the user desires to switch to a new solution series of the same or different problem. Any integral value between one and nine will reset the solution counter to one.

There are a variety of output options which may be selected through the use of IBM 360 Job Control Language[40]. The two principal modes of output communication are 80 column card images and 133 column printed text. In planning the FTXXF001 data sets listed in the system procedure of Figure 24, the user was given a very flexible

list of data sets, numbers FT07 through FT12, upon which to call and use for parameter variation studies. As a result, these later data sets are all 80 bytes long and are blocked at 80 bytes so that the user may employ either cards or directed access devices for data transmission to an appropriate plotter. The NLAP output routine ECB25 is listed in Appendix E. Thus, the user may select the required data set by consulting this subroutine and the definitions of the array names given in Table 3.

A typical NLAP solution obtained without use of the PRINT statement of ECAP is contained in Figure 30 and is the output of the FT06F001 data set.

4.5 Conclusion

The range of validity of the Regression Model has been found to be limited to the active region. The modification of these equations and the subsequent testing were outside the scope of this investigation and are, therefore, a subject for further study.

OUTPUT VARIABLE	NLAP STORAGE ARRAY
Branch Voltage	CCSAV (1, J)
Branch Current	CCSAV (2, J)
Element Voltage	CCSAV (3, J)
Element Current	CCSAV (4, J)
Branch Power Dissipation	CCSAV (5, J)
Node Voltage	CCSAV (6, J)
Transconductance	CDSAV (1, J)
BETA	CDSAV (2, J)
Element Admittance	Y(J)
Element Impedance	X(J)

Table 3. Summary of NLAP Output Variable Storage Arrays

OR EMITTER FOLLOWER, RE = 459.6 OHMS -NLAP

PROBLEM NO. = 2 SOLUTION NO. = 1

BRANCH NUMBER	VOLTAGE SOURCE (VOLTS)	CURRENT SOURCE (MA)	BRANCH RESISTANCE (OHMS)	BRANCH VOLTAGE (VOLTS)	BRANCH CURRENT (MA)	ELEMENT VOLTAGE (VOLTS)	ELEMENT CURRENT (MA)	ELEMENT POWER LOSS (WATTS)	NODE VOLTAGE (VOLTS)	NODE NUMBER
1	1.000	0.0	53.00	-0.996453	0.066918	0.003547	0.066918	0.0000	0.996453	1
2	-0.757	0.0	200.00	0.770775	0.066915	0.013383	0.066915	0.0000	0.225678	2
3	1.150	0.0	0.01	-1.149932	6.745979	0.000067	6.745979	0.0000	1.149932	3
4	2.900	0.0	0.01	-2.899931	6.806821	0.000068	6.806821	0.0000	-2.899931	4
5	0.0	0.0	7500.00	0.924254	6.736044	0.924254	6.736044	0.0062	0.225678	5
6	0.0	0.0	459.60	3.125609	6.800719	3.125609	6.800719	0.0213		
7	0.0	0.0	0.01	0.000000	0.002257	0.000000	0.002257	0.0000		
8	0.0	0.0	100000.00	0.225678	0.002257	0.225678	0.002257	0.0000		

T-NUMBER	ICOLT	IROWT	BETA	GM	YTERM
1	2	5	98.8239746	0.0	0.4941198E 00

RETURN

Figure 30. Typical NLAP 133 Column Solution Printed Output

REFERENCES

1. Brannin, F. H., et al. Computer Aided Design Issue, Proceedings IEEE, Vol. 55, No. 11, Nov. 1967.

2. Kuo, F. F., "Network Analysis and Synthesis", John Wiley and Sons, Inc., New York, 1966, Chapters 6 and 9.

3. Cardenas, Hector, "Design Considerations for High Speed Unsaturated Logic", Unpublished Paper, Texas Instruments, Inc., Dallas.

4. Nanarati, R. P., "An Introduction to Semiconductor Electronics", McGraw-Hill Book Company, Inc., New York, 1963, p.162.

5. Lo, A. W., "Some Thoughts on Digital Components and Circuit Techniques", IRE Transactions on Electronic Components, September, 1961, p.417.

6. Alley, C. L., and Atwood, K. W., "Electronic Engineers", John Wiley and Sons, Inc., New York, 1962, pp.114 and 152.

7. Guth, G., "Unpublished MS Thesis", Department of Computer Science, University of Illinois.

8. Ibid.

9. Hogsett, G. R., "Electronic Circuit Analysis Program ECAP/360-B", TBM Corp., Los Angeles, California, 1966.

10. Littauer, R., "Pulse Electronics", McGraw-Hill Book Company, Inc., New York, 1965, p.316.

11. Greiner, R. A., "Semiconductor Devices and Applications", McGraw-Hill Book Company, Inc., New York, 1961, Chapter 7.

12. Littauer, R., op. cit., pp.17-19.

13. Ryder, R. M., and Kircher, R. M., "Some Circuit Aspects of the Transistor", Bell System Technical Journal, V.28, July, 1949, p.370.

14. Nanavate, R. P., "An Introduction to Semiconductor Electronics", Prentice-Hall, Inc., Englewood Cliffs, N.J., 1963, p.151.

15. DePain, L., "Linear Active Network Theory", Prentice-Hall, Inc., Englewood Cliffs, N.J., 1962, p.65.

16. Brown, J. S., and Bennett, F. C., "The application of Matrices to Vacuum-Tube Circuits", Proceedings IRE, Vol. 36, July, 1948, pp.851-852.

17. Anner, G. E., "Elementary Nonlinear Electronic Circuits", Prentice-Hall, Inc., Englewood Cliffs, N.J., 1967, pp.28-42.

18. Ibid.

19. Golden, J. T., "FORTRAN IV: Programming and Computing", Prentice-Hall, Inc., Englewood Cliffs, N.J., 1965, p.31.

20. Smythe, W. R., Jr., and Johnson, L. A., "Introduction to Linear Programming with Applications", Prentice-Hall, Inc., Englewood Cliffs, N.J., 1966, p.68.

21. Gibbons, J. F., "Semiconductor Electronics", McGraw-Hill Book Company, Inc., New York, 1966, p.405.

22. Gray, P. E., et al, "Physical Electronics and Circuit Models of Transistors", John Wiley and Sons, Inc., New York, 1964, Chapter 9.

23. Malmberg, A. F., et al, "NET-1 Network Analysis Program", Report LA-311, Los Alamos Scientific Laboratory, New Mexico, 1964.

24. Guth, G., op. cit.

25. Ibid.

26. Searle, C. L., "Elementary Circuit Properties of Transistors", John Wiley and Sons, Inc., New York, 1964, p.48.

27. Seshu, S., and Balkonian, N., "Linear Network Analysis", John Wiley and Sons, Inc., New York, 1959, p.68.

28. Crothers, M. H., "Equations for Electrical Network", E.E. 429 Class Notes, University of Illinois, Urbana, Illinois, 1964.

29. Varga, R. S., "Matrix Iterative Analysis", Prentice-Hall, Inc., Englewood Cliffs, N.J., 1962, Chapter 3.

30. Tyson, M., et al, "1620 Electronic Circuit Analysis Program, (ECAP)". User Manual, IBM Report 1620-EE-02X, White Plains, New York, 1965.

31. Guth, G., loc. cit.

32. Hogsett, J. R., "ECAP/360-E, Electronic Circuit Analysis Program", S/360 General Program Library, No. 360D-16.4-4.001, IBM Corporation, Poughkeepsie, New York.

33. Tyson, M., et al, op. cit., p.161.

34. Brannin, F. H., "Computer Methods of Network Analysis", Proceedings of the IEEE, Vol. 55, No. 11, Nov. 1967, p.1787.

35. Millman, J., and Taub, H., "Pulse, Digital and Switching Waveforms", McGraw-Hill Book Company, Inc., New York, 1965, p.217.

36. Guth, G., loc. cit.

37. Tyson, M., et al, "Electronic Circuit Analysis Program (ECAP)", User Manual, IBM Report, 1620-33-02X, IBM Corp., Poughkeepsie, New York, p.2.

38. Hogsett, G. R., op. cit., p.22.

39. Tyson, M., et al, loc. cit., pp.56-65.

40. Editorial Staff, "IBM Operating System/360, FORTRAN IV", IBM Systems Reference Library, File No. S360-25, Form No. C28-6515-3, Programming Systems Staff, IBM Corp., Poughkeepsie, New York, pp.59-61.

41. Editorial Staff, "IBM Operating Systems/360 Job Control Language", IBM Systems Reference Library, File No. S360-48, Form No. C28-6539-3, Programming Systems Staff, IBM Corp., Poughkeepsie, New York.

APPENDIX A

A NETWORK ANALYSIS PROCEDURE

1. Introduction

The formulation of an electronic circuit's mesh solution begins by writing the branch level matrices of Figure A2 for the network shown in Figure A1. Each of these matrices is a column matrix containing one row element for each network branch.

2. Matrix Equation Formulation

The impedance matrix, Z, of Figure A3 is generated using the relation:

$$\text{Matrix element } Z_{ii} = Z_{\text{branch}_i}.$$

The Kirchoff's Voltage Law coefficient matrix, A, of Figure A3 is formed by summing the voltages, V_i, around each mesh with the algebraic sign determined by the agreement (+), or disagreement (-), with the arbitrarily preselected positive current direction. This A matrix has as many rows as the network has meshes and as many columns as the network has branches.

The quantities, Z, JG, E, and A, are processed by a series of matrix equations to arrive at the solution for the network's branch current matrix, I_B. These equations are:

a) All Norton equivalent current sources are converted Thevenin equivalent voltage sources:

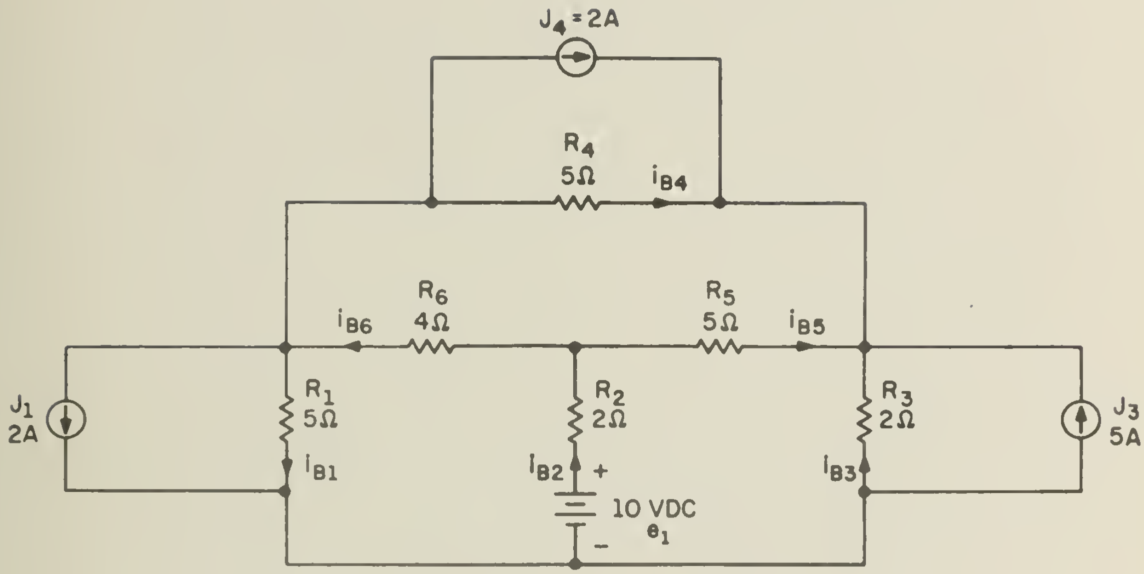


Figure A1. An Equivalent Network

$$E = \begin{bmatrix} 0 \\ +10 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \text{ Volts}$$

$$JG = \begin{bmatrix} +2 \\ 0 \\ +5 \\ +2 \\ 0 \\ 0 \end{bmatrix} \text{ Amps}$$

$$V = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \end{bmatrix} \text{ Volts}$$

$$I = \begin{bmatrix} i_1 \\ i_2 \\ i_3 \\ i_4 \\ i_5 \\ i_6 \end{bmatrix} \text{ Amps}$$

Figure A2. Branch Level E, JG, V, and I Matrices

$$Z = \begin{bmatrix} 5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4 \end{bmatrix} \text{ Amps}$$

$$A = \begin{bmatrix} -1 & -1 & 0 & 0 & 0 & -1 \\ 0 & +1 & -1 & 0 & +1 & 0 \\ 0 & 0 & 0 & +1 & -1 & +1 \end{bmatrix}$$

Figure A3. Branch Level Z and A Matrices

$$N = Z * JG$$

b) The converted current sources are summed with the native Thevenin equivalent voltage sources:

$$EE = E + N.$$

c) The voltage sources around each mesh are summed by a matrix equation:

$$EM = A * EE.$$

d) The mesh impedance matrix is formed by transposing the A matrix, premultiplying the Z matrix by A, and then post-multiplying A * Z, the A * Z product by A^T:

$$AM = A * Z * A^T.$$

e) The network's mesh current is calculated by premultiplying the EM matrix by the inverse of the ZM matrix:

$$IM = YM * EM.$$

f) The final step is to obtain the individual currents in each branch by summing up the mesh currents passing through each branch and then subtracting the current generator matrix, JG:

$$IB = A^T * IM - JG.$$

3. <u>Evaluation of Matrix Equations</u>

The above matrix equations are herein evaluated for the network of Figure A1.

a)

$$N = Z * JG = \begin{bmatrix} 5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4 \end{bmatrix} * \begin{bmatrix} 2 \\ 0 \\ 5 \\ 2 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 10 \\ 0 \\ 10 \\ 10 \\ 0 \\ 0 \end{bmatrix} \text{ Volts}$$

b)

$$EE = E + N = \begin{bmatrix} 0 \\ 10 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 10 \\ 0 \\ 10 \\ 10 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 10 \\ 10 \\ 10 \\ 10 \\ 0 \\ 0 \end{bmatrix} \text{ Volts}$$

c)

$$EM = \begin{bmatrix} -1 & -1 & 0 & 0 & 0 & -1 \\ 0 & +1 & -1 & 0 & +1 & 0 \\ 0 & 0 & 0 & +1 & -1 & +1 \end{bmatrix} * \begin{bmatrix} 10 \\ 10 \\ 10 \\ 10 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} -20 \\ 0 \\ 10 \end{bmatrix} \text{ Volts}$$

d)

$$AM = [A*Z*A^T] = \begin{bmatrix} -1 & -1 & 0 & 0 & 0 & -1 \\ 0 & +1 & -1 & 0 & +1 & 0 \\ 0 & 0 & 0 & +1 & -1 & +1 \end{bmatrix} * \begin{bmatrix} 5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4 \end{bmatrix} * \begin{bmatrix} -1 & 0 & 0 \\ -1 & +1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & +1 \\ 0 & +1 & -1 \\ -1 & 0 & +1 \end{bmatrix} = \begin{bmatrix} 11 & -2 & -4 \\ -2 & +9 & -5 \\ -4 & -5 & 14 \end{bmatrix} \text{ ohms}$$

e)

$$IM = \begin{bmatrix} IM_1 \\ IM_2 \\ IM_3 \end{bmatrix} = YM * EM = \frac{10}{831} \begin{bmatrix} +101 & -48 & -46 \\ -48 & +138 & -63 \\ +46 & -63 & +95 \end{bmatrix} \begin{bmatrix} -2 \\ 0 \\ +1 \end{bmatrix} = \frac{1}{831} \begin{bmatrix} -156 \\ -33 \\ +3 \end{bmatrix} \begin{bmatrix} -1.88 \\ -0.397 \\ -0.0367 \end{bmatrix} \text{ amps}$$

f)

$$IB = A^T * IM - JG = \begin{bmatrix} -1 & 0 & 0 \\ -1 & +1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & +1 \\ 0 & +1 & -1 \\ -1 & 0 & +1 \end{bmatrix} \begin{bmatrix} -1.88 \\ -0.397 \\ +0.036 \end{bmatrix} - [J] = \begin{bmatrix} +1.88 \\ +1.48 \\ +0.40 \\ +0.036 \\ -0.44 \\ +1.42 \end{bmatrix} = \begin{bmatrix} -0.12 \\ +1.48 \\ +4.60 \\ -1.96 \\ -0.44 \\ +1.92 \end{bmatrix} \text{ amps}$$

A network analysis computer program utilizing the above matrix equations is listed in Figure A4, with a sample output shown in Figure A5.

```
FLOAT;
ARRAY A 3 6 F; ARRAY E 6 1 F;
ARRAY JG 6 1 F; ARRAY Z 6 6 F;
ARRAY N 6 1 F; ARRAY EE 6 1 F;
ARRAY EM 3 1 F; ARRAY AT 6 3 F;
ARRAY ZAT 6 3 F; ARRAY AM 3 3 F;
ARRAY YM 3 3 F; ARRAY IM 3 1 F;
ARRAY IBM 6 1 F; ARRAY IB 6 1 F;
ARRAY IBX 6 6 F; ARRAY IBS 6 1 F;
ARRAY PD 6 1 F; ARRAY P 6 6 F;
ARRAY RP 6 6 F; ARRAY S 6 1 F;
ARRAY SP 6 1 F;
A[0,0]=-1; A[0,1]=-1; A[0,5]=-1;
A[1,1]= 1; A[1,2]=-1; A[1,4]= 1;
A[2,3]= 1; A[2,4]=-1; A[2,5]= 1;
E[1,0]=10;
JG[0,0]= 2; JG[2,0]= 5; JG[3,0]= 2;
RP[0,0]= 1; RP[1,1]= 1; RP[2,2]= 1;
Z[0,0]= 5; Z[1,1]= 2; Z[2,2]= 2; Z[3,3]= 5;
A[3,3]= 5; Z[4,4]= 5; Z[5,5]= 4;
N = Z * JG; EE = E + N; EM = A * EE;
FOR I = 0, 1, 2;  BEGIN FOR K = 0, 1, 5;  BEGIN
AT[K, I] = A[I, K];  END   END
ZAT = Z * AT; AM = A * ZAT; YM = ZM** -1.0;
IM = YM * EM; IBM = AT * IM; IB = IBM - JG;
IBX[0,0] = IB[0,0];
IBX[3,3] = IB[3,0];
IBS = IBX * IB;
PD = Z * IBS; P = RP** -1.0;
S = P * PD; S = 100 *S;
FORMAT LINE, S27, C*MATCK DATA OUTPUT*//
PRINT LINE;
FORMAT HDR, C*     I     Z     IB     PD     RP
          S          SP*/;
PRINT HDR;
FORMAT UNITS, C* BRANCH OHMS   AMPS   WATTS
    WATTS       NONE      PERCENT*/;
PRINT UNITS;
FORMAT MATCK, I6 ,6(Flo.2);
FOR  I = 0,1,5;  BEGIN
PRINT MATCK,  I,Z[I,I],IB[I,0],PD[I,0],RP[I,I],5[I,0],SP[I,0];
END;
```

Figure A4. Network Analysis Program

MATCK DATA OUTPUT

I	Z	IB	PD	RP	S	SP
BRANCH	OHMS	AMPS	WATTS	WATTS	NONE	PERCENT
0	5.00	-0.12	0.08	1.00	0.08	7.53
1	2.00	1.48	4.38	1.00	4.38	438.17
2	2.00	-4.60	42.37	1.00	42.37	4237.32
3	5.00	-1.96	19.28	1.00	19.28	1928.45
4	5.00	-0.43	0.94	1.00	0.94	93.84
5	4.00	1.91	14.64	1.00	14.64	1464.37

Figure A5. Output of Network Analysis Test Program

APPENDIX B

REGRESSION MODEL SUBROUTINE

1 SUBROUTINE ITERATE LISTING

```
      COMMON CCSAV(6,200),IDWORD(74)
      DOUBLE PRECISION ZPRL(50,50)                                    GNL00030
      COMMON NMAX,NNODE,NTERMS,NUMBL,NUMBR,NUMBC,IRTN,NTRACE,NSWTCH,KTO,GNL00050
     1 NPRINT(10)                                                     GNL00060
      COMMON E(200),EMIN(200),EMAX(200),AMP(200),AMPMIN(200),AMPMAX(200)GNL00070
      COMMON Y(200),YMIN(200),YMAX(200),NINIT(200),NFIN(200),MODE1(200) GNL00080
      COMMON YTERM(200),YTERMH(200),YTERML(200),IROWT(200),ICOLT(200)   GNL00090
      COMMON ERROR1,ISEQ,MSEQ,MO,NUMMO,VFIRST(50),VSECND(50),VLAST(50)  GNL00100
      COMMON MOBRN(50),MOPARM(50),MOSTEP(50),IWCOUT(4)                  GNL00110
C                                                                       GNL00120
C     THE FOLLOWING VARIABLES ARE USED ONLY IN THE ECAP D.C. ANALYSIS   GNL00130
      COMMON AX1,SMLEP(50),CURR(200),SMLE(200),EQUCUR(50),EX(200)       GNL00150
      COMMON EB(200),AMPX(200),AMPB(200),VNOM(50),STDSQ(50),L,M,IITOL   GNL00160
      COMMON JX1,JX4,JX5,DELTA,DUM1(28)                                 GNL00170
      COMMON NWORDS(72),NMCD(2,20),KLABEL(4),KPUNC(5),INDC(2,20)        GNL00190
      COMMON INPUTB(9),NBCD(20),KTYPE(5),NBLANK,NOEXEC,ITOL,NEQUIM,IPC  GNL00191
      COMMON INVAL,LL,ICOL,LTYPE,KCCL,NQUIT,ITRANS,KO,KS,KELAST,NUM,M1  GNL00192
      COMMON M2,M3,KCARD,KG,NP,NTR,MAC,HNODE,TNUM,NOEL,NOE,NOI,NOIC     GNL00193
      COMMON EQUIVN(20),KOUT(2,10)                                      GNL00194
      COMMON MATA(200,4,3),YX(200),YB(200),YTERMX(200),YTERMB(200)      GNL00210
      COMMON WCMAX(50),WCMIN(50)                                        GNL00220
      COMMON  CCSAV(2,200)
      DOUBLE PRECISION SMLEP,CURR,SMLE,EQUCUR                           GNL00240
C                                                                       GNL00270
C     THE FOLLOWING VARIABLES ARE USED ONLY IN SUBROUTINE ITRATE
      DIMENSION NITER(6,2),COEFS(50,5),ICRES(50),IPTYPE(50),IPLCC(50)   GNL00280
      DIMENSION ICURR(25),TOL(25),AMPOLD(25),BETA(200),GM(200)          GNL00290
      DOUBLE PRECISION MACUR(200)                                       WNL00295
      REAL*8  IREF                                                      WNL00297
C
C     INITIALIZE AND DEFINE DICTIONARY AND INDICATORS                   GNL00310
      IF (NTRACE) 1,3,1                                                 GNL00330
    1 WRITE (6,2)                                                       GNL00340
    2 FORMAT (' SUBROUTINE ITRATE, REGRESSION MODEL VERSION ENTERED')
    3 NITER(1,1)=   -482328512                                          GNL00360
      NITER(1,2)=   -985644992                                          GNL00370
      NITER(2,1)=  -1019199424                                          GNL00380
      NITER(2,2)=   -465551296                                          GNL00390
      NITER(3,1)=   -482328512                                          GNL00400
      NITER(3,2)=   -700432320                                          GNL00410
      NITER(4,1)=   -985644992                                          GNL00420
      NITER(4,2)=   -415219648                                          GNL00430
      NITER(5,1)=  -733986752
```

```
      NITER(5,2)=  -700432320
      NITER(6,1)=  -650100672
      NITER(6,2)=  -985644992                                           GNL00440
      DO 300 I=1,5C                                                     GNL00450
      ICRES(I)=0                                                        GNL00460
      IPTYPE(I)=0                                                       GNL00470
      IPLOC(I)=0                                                        GNL00480
      DO 300 J=1,5                                                      GNL00490
  300 COEFS(I,J)=0.0                                                    GNL00500
      DO 301 I=1,25                                                     GNL00510
      ICURR(I)=0                                                        GNL00520
      TOL(I)=0.0                                                        GNL00530
  301 AMPOLD(I)=0.0                                                     WNL00532
      DO 302 I=1,200                                                    WNL00534
      BETA(I)=0.0                                                       WNL00536
      MACUR(I) = 0.0                                                    WNL00538
  302 GM(I)=0.0                                                         GNL00540
      INDCUR=0                                                          GNL00550
      INDTOL=0                                                          GNL00560
      INDTEM=0                                                          GNL00570
      TEMP=0.0                                                          GNL00580
      TEMPI=0.0                                                         GNL00590
      TEMPF=0.0                                                         GNL00600
      TEMPC=0.0                                                         GNL00610
      IINUM=0                                                           GNL00620
      ITEMP2=1                                                          GNL00630
C                                                                       GNL00640
C     READ AND LIST DATA CARD                                           GNL00660
 1000 READ (5,4) NWORDS                                                 GNL00670
    4 FORMAT (72A1)                                                     GNL00680
      WRITE (6,5) NWORDS                                                GNL00690
    5 FORMAT (1X,72A1)                                                  GNL00700
      KCARD=KCARD+1                                                     GNL00710
      NOEXEC=NOEXEC+NQUIT
C
C     CHECK FOR 'IC' CARD
      IF (NWORDS(1)-INDC(1,15)) 1004,1001,1004
 1001 IF (NWORDS(2)-NMCD(1,1)) 1004,1C02,1004
 1002 DO 1003 ICOL=1,72
 1003 IDWORD(ICOL)=NWORDS(ICOL)
      GO TO 100C
C
 1004 CONTINUE                                                          GNL00720
      IF (NWORDS(1)-INPUTB(8)) 2000,100C,2000                           GNL00730
C                                                                       GNL00740
C     SUPPRESS BLANKS FROM COLUMNS 7-72                                 GNL00750
C                                                                       GNL00760
 2000 KCOL=6                                                            GNL00770
      DO 7 ICOL=7,72                                                    GNL00780
      IF (NWORDS(ICOL)-NBLANK) 6,7,6                                    GNL00790
    6 KCOL=KCOL+1                                                       GNL00800
      NWORDS(KCOL)=NWORDS(ICOL)                                         GNL00810
    7 CONTINUE                                                          GNL00820
C                                                                       GNL00830
C     CHECK CONTENTS OF COLUMNS 1-6                                     GNL00840
C                                                                       GNL00850
      CO 11 ICOL=1,5                                                    GNL00860
      IF (NWORDS(ICOL)-NBLANK) 8,11,8                                   GNL00870
    8 DO 9 LTYPE=1,2
```

```
      IF (NWORDS(ICOL)-KLABEL(LTYPE)) 9,200,9
    9 CONTINUE
      M3=5
   15 ITRANS=6
   10 WRITE (6,3006) M3
      WRITE (6,3007) KCARD,ICOL
      WRITE (6,3008)
      NQUIT=1
 3006 FORMAT ('0 ***** ERROR NO.',I3,' **** MSG FROM SUBROUTINE ITRATE')
 3007 FORMAT ('   CARD NO.=',I3,3X,  'APPROXIMATE COLUMN NO. IS',I3,
     1'MSG FROM SUBROUTINE ITRATE')
 3008 FORMAT (//)
      GO TO 1000
   16 FORMAT ('0','SUBROUTINE ITRATE ERROR MSG - CHECK YOUR INPUT DATA O
     1VER CAREFULLY, PGM SAYS YOU HAVE MADE A GOOF')
   11 CONTINUE
      IF (KCOL-6) 500,12,13
   12 M3=4
      GO TO 15
  500 ITRANS=5
      GO TO 10
C
C     CHECK FOR TEMPERATURE, CURRENT, TOLERANCE, EXECUTE, MODIFY OR
C     RETURN STATEMENT,  INDEX NUMBERS 1, 2, 3, 4, 5, 6 RESPECTIVELY.
C
   13 ICOL=7
      DO 20 IDENT=1,6
      IF (NWORDS(ICOL)-NITER(IDENT,1)) 20,14,20
   14 IF (NWORDS(ICOL+1)-NITER(IDENT,2)) 20,25,20
   20 CONTINUE
      M3=12
      GO TO 15
   21 M3=34
      GO TO 15
   25 GO TO (26,26,26,8000,4000,9999),IDENT
C
C     GET PAST '=' SIGN ON TOLERANCE, TEMPERATURE OR CURRENTS
C
   26 ICOL=ICOL+1
      IF (ICOL-KCOL) 27,21,21
   27 IF (NWORDS(ICOL)-KPUNC(1)) 26,28,26
   28 ICOL=ICOL+1
      GO TO (50,29,39),IDENT
C
C     PROCESS TOLERANCES AND/OR CURRENTS
C
   29 IF (INDCUR) 21,30,21
   30 INDCUR=1
   41 I=0
   31 CALL EC809
      I=I+1
      NUM=TNUM
      IF (I-25) 42,42,21
   32 ICURR(I)=NUM
      IT=I
   33 ICOL=ICOL+1
      IF (ICOL-KCOL) 34,34,1000
   34 IF (NWORDS(ICOL)-KPUNC(3)) 36,35,36
   35 ICOL=ICOL+1
```

```
      GO TO 31
   36 M3=34
      GO TO 15
   39 IF (INDTOL) 21,40,21
   40 INDTOL=1
      GO TO 41
   42 GO TO (500,32,43),IDENT
   43 TOL(I)=TNUM
      JT=1
      GO TO 33
C
C     PROCESS TEMPERATURE
C
   50 IF (ICOL-KCOL) 51,51,36
   51 IF (INDTEM) 36,52,36
   52 INDTEM=1
      CALL ECB09
      TEMPI=TNUM
      ICOL=ICOL+1
      IF (ICOL-KCOL) 54,54,53
   53 TEMP=TEMPI+273.0
      ITEMP2=1
      GO TO 1000
   54 IF (NWORDS(ICOL)-KPUNC(2)) 36,55,36
   55 ICOL=ICOL+1
      CALL ECB09
      TEMPC=TNUM
      ITEMP2=2
      ICOL=ICOL+1
      IF (ICOL-KCCL) 56,56,36
   56 IF (NWORDS(ICOL)-KPUNC(4)) 36,57,36
   57 ICOL=ICOL+1
      CALL ECB09
      TEMPF=TNUM
      GO TO 1000
C
C     GET COEFFICIENTS AND DATA TYPE
C
  200 ICOL=ICOL+1
      IF (ICOL-KCOL) 201,201,21
  201 IINUM=IINUM+1
      IF (IINUM-50) 203,203,202
  202 M3=36
      GO TO 15
  203 CALL ECB09
      NUM=TNUM
      IPLOC(IINUM)=NUM
      IF (NWORDS(7)-INPUTB(5)) 204,207,204
  204 IF (NWORDS(7)-INPUTB(2)) 205,208,205
  205 IF (NWORDS(7)-INPUTB(4)) 206,209,206
  206 IF (NWORDS(7)-INPUTB(3)) 225,224,225
  207 MTYPE=1
      GO TO 210
  208 MTYPE=2
      GO TO 210
  209 MTYPE=3
  210 IPTYPE(IINUM)=MTYPE
  211 ICOL=ICOL+1
      IF (ICOL-KCCL) 212,212,21
```

```
  212 IF (NWORDS(ICOL)-KPUNC(2)) 211,213,211                          GNL01970
  213 ICOL=ICOL+1                                                      GNL01980
      IF (ICOL-KCOL) 214,214,21                                        GNL01990
  214 CALL ECB09                                                       GNL02000
      NUM=TNUM                                                         GNL02010
      ICRES(IINUM)=NUM                                                 GNL02020
      ICOL=ICOL+1                                                      GNL02030
      IF (ICOL-KCOL) 215,215,21                                        GNL02040
  215 IF (NWORDS(ICOL)-KPUNC(4)) 206,216,206                          GNL02050
  216 ICOL=ICOL+1                                                      GNL02060
      IF (ICOL-KCOL) 217,217,21                                        GNL02070
  217 IF (NWORDS(ICOL)-KPUNC(1)) 206,218,206                          GNL02080
  218 IICOL=0                                                          GNL02090
  222 ICOL=ICOL+1                                                      GNL02100
      IF (ICOL-KCOL) 220,220,219                                       GNL02110
  219 IF (IICOL) 500,206,1000                                          GNL02120
  220 IICOL=IICOL+1                                                    GNL02130
      IF (IICOL-5) 223,223,206                                         GNL02140
  223 CALL ECB09                                                       GNL02150
      COEFS(IINUM,IICOL)=TNUM                                          GNL02160
      ICOL=ICOL+1                                                      GNL02170
      IF (ICOL-KCOL) 221,221,1000                                      GNL02180
  221 IF (NWORDS(ICOL)-KPUNC(3)) 206,222,206                          GNL02190
  224 MTYPE=4                                                          GNL02200
      GO TO 210                                                        GNL02210
  225 M3=36                                                            GNL02220
      GO TO 15                                                         GNL02230
C                                                                      GNL02240
C     BEGIN EXECUTION OF ITERATION TECHNIQUE                           GNL02250
C                                                                      GNL02260
 8000 IF (NTRACE) 8901,8900,8901                                       GNL02270
 8901 WRITE (6,8908)                                                   GNL02280
      WRITE (6,8902) (TOL(I),I=1,IT)                                   GNL02290
      WRITE (6,8903) (ICURR(I),I=1,IT)                                 GNL02300
      WRITE (6,8906)                                                   GNL02310
      WRITE (6,8904) ((COEFS(I,J),J=1,5),I=1,IINUM)                    GNL02320
      WRITE (6,8907)                                                   GNL02330
      WRITE (6,8905) (I,ICRES(I),IPTYPE(I),IPLOC(I),I=1,20)            GNL02340
      WRITE (6,8911)                                                   GNL02350
      WRITE (6,8910) (CURR(J),J=1,NMAX)                                GNL02360
 8902 FORMAT (1H0,10F10.4)                                             GNL02370
 8903 FORMAT (1H0,10(5X,I5))                                           GNL02380
 8904 FORMAT (1X,5E17.7)                                               GNL02390
 8905 FORMAT (1X,I3,5X,I5,5X,I5,5X,I5)                                 GNL02400
 8906 FORMAT (1H0,12HCOEFFICIENTS/)                                    GNL02410
 8907 FORMAT (1H0,2X,1HI,7X,5HICRES,4X,6HIPTYPE,5X,5HIPLOC/)           GNL02420
 8908 FORMAT (1H0,23HTOLERANCES AND CURRENTS)                          GNL02430
 8910 FORMAT (1X,8E16.7)                                               GNL02450
 8911 FORMAT (1H0,8HCURRENTS/)                                         GNL02460
 8900 IF (INDCUR+INDTOL-2) 8001,8002,8001                              GNL02480
 8001 M3=35                                                            GNL02490
      GO TO 15                                                         GNL02500
 8002 IF (IT-JT) 8003,8004,8003                                        GNL02510
 8003 M3=36                                                            GNL02520
      GO TO 15                                                         GNL02530
 8004 CONTINUE                                                         GNL02540
      NUM=0                                                            GNL02550
      GO TO (8098,8050),ITEMP2                                         GNL02560
C                                                                      GNL02570
```

```
C     SINGLE TEMPERATURE                                                GNL02580
C                                                                       GNL02590
 8008 CONTINUE                                                          GNL02600
      GO TO 1000
C                                                                       GNL02620
C     RANGE OF TEMPERATURES                                             GNL02630
C                                                                       GNL02640
 8050 TEMPC=(TEMPF-TEMPI)/TEMPC                                         GNL02650
      TEMPF=TEMPF+273.0                                                 GNL02660
      TEMPI=TEMPI+273.0                                                 GNL02670
      TEMP=TEMPI                                                        GNL02680
      GO TO 8098                                                        GNL02690
 8053 TEMP=TEMP+TEMPC                                                   GNL02700
      NUM=0                                                             GNL02710
      IF (TEMP-TEMPF) 8098,8098,8054                                    GNL02720
 8054 CONTINUE                                                          GNL02730
      GO TO 100C
C                                                                       GNL02750
C     STORE PRESENT CURRENT VALUES                                      GNL02760
C                                                                       GNL02770
 8098 DO 8099 I=1,IT                                                    GNL02780
      J=ICURR(I)                                                        GNL02790
 8099 AMPOLD(I)=CURR(J)                                                 GNL02800
      NUM=NUM+1                                                         GNL02810
C                                                                       GNL02820
C     CALCULATE NEW DEPENDENT VALUES                                    GNL02830
C
 8100 DO 8105 I=1, IINUM                                                WNL02850
      J=ICRES(I)                                                        WNL02851
      IREF = 10COCCC.C
      CUTOFF = 1.0
      ZOFF = 5555555.CCCO
      MACUR(J) = (CURR(J))*IREF
      SAVE = SNGL(MACUR(J))                                             WNL
      MTYPE=IPTYPE(I)                                                   WNL02853
      K=IPLOC(I)                                                        WNL02854
      NFBR = ICOLT(K)                                                   WNL02855
C                                                                       WNL02856
      GOTO (8101,8102,8103,8104),MTYPE                                  WNL02860
C                                                                       WNL02861
C   CALCULATE NEW VBE
 8101 IF(SAVE .LE. CUTOFF) WRITE(6,8920) J,SAVE,J                       WNL028X
 8920 FORMAT(' ',20X,'CALCULATION OF VBE',      5X ,' CURR(',I3,') =' ,
     1 F20.8, 5X,'FROM BRANCH =',I3)
      IF(SAVE .LE. CUTOFF) GOTO 614
      ALCUR = ALOG(SAVE)                                                WNL02870
      VALUE=COEFS(I,1) + ALCUR*(COEFS(I,2)+ COEFS(I,3)*ALCUR)           WNL02872
      E(K)  = VALUE   + CCSAV(3,K)                                      WNL0287
      GOTO 616                                                          WNL02873
  614 VALUE = COEFS(I,1)                                                WNL02875
      E(K)  = VALUE                                                     WNL02876
  616 CONTINUE
C 616 WRITE(6,8928)J,SAVE,J,CURR(J),MTYPE,K,NFBR,ALCUR,VALUE
 8928 FORMAT('0',10X,'CALCULATION OF VBE',5X,'J='I3,5X,' IC =',F8.3,
     15X,'CURRN('I3,')= 'F12.6,5X,'MTYPE ='I3,5X,'K ='I3,/
     2' ',34X,'NFBR ='I3,5X,'LN(IC) ='F12.6,5X,' VBE ='F12.6/ )
      GOTO 8105
C
C                                                                       WNL02878
```

```
C  CALCULATE NEW BETA
C
 8102 IF(SAVE .LE. CUTOFF) WRITE(6,8922) J,SAVE,J                          WNL02880
 8922 FORMAT(' ',20X,'CALCULATION OF BETA',      5X ,' CURR(',I3,') =' ,
     1 F20.8, 5X,'FROM BRANCH =',I3)
      IF(SAVE .LE. CUTOFF) GOTO 624
      ALCUR = ALOG(SAVE)                                                   WNL02883
      VALUE = COEFS(I,1) + ALCUR*(COEFS(I,2) + COEFS(I,3)*ALCUR)          WNL02884
      Y(NFBR) = 1.0/(ZOFF)                                                 WNL02887
      BETA(K) = VALUE                                                      WNL0288
      GOTO 626                                                             WNL02885
  624 VALUE = 0.0                                                          WNL02886
      BETA(K) = VALUE                                                      WNL02888
      GOTO 626                                                             WNL02889
  626 YTERM(K) = VALUE * Y(NFBR)                                           WNL02892
C     WRITE(6,8929)J,SAVE,J,CURR(J),MTYPE,K,NFBR,ALCUR,VALUE,K,YTERM(K)
 8929 FORMAT('0',10X,'CALCULATION OF BETA',5X,'J=',I5,5X,' IC = 'F8.3,
     15X,'CURRN('I3,')= 'F12.6,5X,'MTYPE ='I3,5X,'K ='I3,/
     2' ',34X,'NFBR ='I3,5X,'LN(IC) ='F12.6,5X,'BETA ='F12.6/
     3' ',33X,'YTERM('I2,') =',F12.6/)
      GOTO 8105                                                            WNL02896
C
C
C  CALCULATE NEW VALUE OF GM
C
 8103 IF(SAVE .LE. CUTOFF) WRITE(6,8924) J,SAVE,J
 8924 FORMAT(' ',15X,'CALCULATION OF GM  ',/'0',20X ,'ICURR(',I3,') ='
     1 F12.6,10X,'FROM BRANCH =',I3)
      VALUE=COEFS(I,I) + ALCUR*(COEFS(I,2)+ COEFS(I,3)*ALCUR)             WNL03012
      GM( K)=VALUE                                                         WNL03014
      YTERM( K)=VALUE                                                      WNL03016
      GO TO 8105                                                           GNL03030
C
C
C  CALCULATE NEW VALUE OF  Y
C
 8104 IF(SAVE .LE. CUTOFF) WRITE(6,8926) J,SAVE,J
 8926 FORMAT(' ',15X,'CALCULATION OF  Y  ',/'0',20X ,'ICURR(',I3,') ='
     I F12.6,10X,'FROM BRANCH =',I3)
      VALUE=COEFS(I,I) + ALCUR*(COEFS(I,2)+ COEFS(I,3)*ALCUR)             WNL03042
      Y(NFBR)=1.0/VALUE                                                    WNL03044
 8105 CONTINUE                                                             GNL03050
C
C
C     SOLVE DC PROBLEM                                                     GNL03070
C                                                                          GNL03080
      DO 8106 I=1,NMAX                                                     GNL03110
      YX(I)=Y(I)                                                           GNL03120
      EX(I)=E(I)                                                           GNL03130
 8106 AMPX(I)=AMP(I)                                                       GNL03140
      IF (NTERMS) 8107,8109,8107                                           GNL03150
 8107 DO 8108 I=1,NTERMS                                                   GNL03160
 8108 YTERMX(I)=YTERM(I)                                                   GNL03170
 8109 CALL ECB22(ZPRL)                                                     GNL03180
      CALL ECB23                                                           GNL03190
      CALL ECB26(ZPRL)                                                     GNL03200
      CALL ECB25                                                           GNL03210
      IF (NUM-20) 8200,8300,8200                                           GNL03230
C                                                                          GNL03240
```

```
C      CHECK TOLERANCES                                                  GNL03250
C                                                                        GNL03260
 8200 DO 8201 I=1,IT                                                     GNL03270
      J=ICURR(I)                                                         GNL03280
      SAVE=CURR(J)
      IF (ABS(AMPOLD(I)-SAVE) - ABS(AMPOLD(I)*TOL(I))) 8201,8201,8098
 8201 CONTINUE                                                           GNL03300
C                                                                        GNL03320
C      OUTPUT RESULTS                                                    GNL03330
C                                                                        GNL03340
 8302 WRITE (6,8303) NUM                                                 GNL03350
 8303 FORMAT ('-', 'NUMBER OF ITERATIONS = ',I3/)                        WNL03360
      IF (INDTEM) 8202,8203,8202                                         GNL03370
 8202 TEMPA=TEMP-273.0                                                   GNL03380
      WRITE (6,9000) TEMPA                                               GNL03390
 9000 FORMAT (1H0,14HTEMPERATURE = ,F10.3/)                              GNL03400
 8203 DO 8204 I=1,NMAX                                                   GNL03410
      YX(I)=Y(I)                                                         GNL03420
      EX(I)=E(I)                                                         GNL03430
 8204 AMPX(I)=AMP(I)                                                     GNL03440
      IF (NTERMS) 8205,82C7,8205                                         GNL03450
 8205 DO 8206 I=1,NTERMS                                                 GNL03460
 8206 YTERMX(I)=YTERM(I)                                                 GNL03470
 8207 CALL ECB22(ZPRL)                                                   GNL03480
      CALL ECB23                                                         GNL03490
C    THE FOLLOWING CODE RECOGNIZES AND EXECUTES THE PRINT NV,MI CONTROL
C    INPUT DATA STATEMENT. RESULTANT OUTPUT IS THE DC AND NL MATRICES.
      IF(NPRINT(10)) 1C7,107,108
  108 JX1=1
      CALL ECB24(ZPRL)
  107 CALL ECB26(ZPRL)
      NTR=1
      IF(NPRINT(1C)) 1C3,103,104
  104 JX1=2
      CALL ECB24(ZPRL)
  103 CALL ECB25
      NTR=4
      WRITE (6,9001)                                                     GNL03530
      WRITE (6,9002) (I,ICCLT(I),IROWT(I),BETA(I),GM(I),YTERM(I),        GNL03540
     1                   I=1,NTERMS)                                     GNL03550
 9001 FORMAT (//,1X,9H T-NUMBER,7X,6H ICOLT,11X,6H IROWT,6X,4HBETA,11X,  GNL03580
     12HGM,11X,5HYTERM/)                                                 GNL03590
 9002 FORMAT (3X,I4,11X,I4,13X,I4,2F15.7,E15.7)                          GNL03600
      GO TO (80C8,8C53),ITEMP2                                           GNL03630
 8300 WRITE (6,83C1)                                                     GNL03640
 8301 FORMAT (1HC,34HNO CCNVERGENCE AFTER 20 ITERATIONS/)                GNL03650
      GO TO 8302                                                         GNL03660
C
C      BEGIN MODIFY ROUTINE
C
 4000 CONTINUE
      GO TO 1000
C
C      EXIT TO MAIN ROUTINE
C
 9999 RETURN
      END                                                                GNL03670
```

```
2  COMMENTS ON PROGRAM CHANGES

C         1. ORIGINAL VERSION OF THIS SUBROUTINE WAS WRITTEN BY G. GUTH.
C            CHANGES MADE BY D. WOOD IN INPUT-OUTPUT ROUTINE TO PERMIT
C            PARAMETER VARIATION ANALYSIS, USE OF VARYING  REFERENCE
C            CURRENTS, IREF, AND MODIFICATION  OF THE BASIC ROUTINE TO
C            PERMIT IT TO HANDLE BOTH CUTOFF AND ACTIVE TRANSISTOR
C            OPERTAING REGIONS.
C
C         2. INSERTED MACUR(J) TO IMPROVE THE FLEXIBILTY OF REGRESSION
C            MODEL.  THUS BY THE USE OF A SCALE FACTOR OF IO = IREF, THE
C            CURRENT NORMALIZATION INTHE REGRESSION MODEL MAY BE CHANGED
C            BY A FUTURE MODIFICATION PERMITTING IREF TO BE READ IN AT
C            EXECUTION TIME.
C
C         3. CUTOFF IS THE COLLECTOR CURRENT VALUE, IC,  AT WHICH BETA IS
C            SET EQUAL TO ZERO. CUTOFF IC IS THAT VALUE OF IC, DETERMINED
C            FROM THE REGRESSION MODEL EQUATIONS, FOR WHICH BETA = 0.
C
C         4. ZOFF IS THE IMPEDANCE LEVEL SWITCHED INTO THE BASE BRANCH
C            WHENEVER THE COLLECTOR CURRENT BECOMES .LE. CUTOFF.
C            ZOFF SET EQUAL TO = 5555555.0 IN PROGRAM, THUS ENABLING THE
C            USER TO SPOT IN THE IMPEDANCE LISTINGS, EXACTLY WHEN THE
C            PROGRAM PREDICTED ENTRY INTO THE CUTOFF MODE OF OPERATION.
C
C         5. DUE TO THE THE UNIT OF CURRENT, AMPERES, ASSUMED BY THE ECAP
C            SUBROUTINES, SET IREF=1.0, WHEN A 1.0 AMPERE REFERENCE IS
C            DESIRED. IF MICROAMPERES IS DESIRED, SET IREF=1000000.0, AND
C            IF MILLIAMPERES IS DESIRED, SET IREF=1000.0.
C
C         6. THE ARRAY IN COMMON, CDSAV(2,200) IS USED TO PASS YTERM(200)
C            AND  BETA(200) VALUES TO NLAP OUTPUT ROUTINE ECB25.
C            CDSAV(1,J) = YTERM(J) AND CDSAV(2,K) = BETA(K).
C
C         7. THE FOLLOWING INFORMATION IS TO ASSIST PGMR IN ADDING MODIFY
C            OPTION TO THIS SUBROUTINE, TO ELIMINATE PRESENT REQUIREMENT
C            FOR DUPLICATING DATA DECKS FOR THE CASE OF PARAMETER
C            VARIATION OPTION RUNS.
C              A. MODIFY ROUTINE IS ECA30
C              B. M = MODIFY NO., NUMMO= NO. MODS, MOSTEP= NO. MOD STEPS
C              C. MOPARM(M) IS KIND OF PARAMETER TO BE MODIFIED
C              D. MOBRN(M) IS CARD SEQUENCE NUMBER
C              E. NTERM IS TOTAL NUMBER OF DATA CARDS
C              F. VFIRST(M)  CONTAINS VALUE OF ALL PARMS TO BE MODIFIED
C
C         8. NOTE THAT GM AND Y NEW VALUE CODES HAVE NOT  BEEN MODIFIED.
C
C         9. THIS SUBROUTINE AS NOW CODED WILL GIVE ACCURATE RESULTS FOR
C            THE NL OPTION OF NLAP, PROVIDED THAT THE USER IS VERY
C            CAREFUL TO KEEP INPUT DATA SO THAT THE CIRCUIT DOES NOT GO
C            DEEP INTO THE CUTOFF REGION OF OPERATION, OTHERWISE CURRENT
C            CONVERGENCE MAY NOT BE OBTAINED DUE TO  PRESENT REGRESSION
C            MODEL EQUATION FOR VBE NOT HAVING AN EXTERNAL CIRCUIT
C            CURRENT VARIABLE INCLUDED AS AN INDEPENDENT VARIABLE.
```

APPENDIX C

LINEAR REGRESSION PROGRAM

1 LINEAR REGRESSION PROGRAM LISTING

```
      COMMON   A,ACHG,DLD,ICDCT
      COMMON AI(40),ANAM(25,40),FMT(18),KZ(20),KCHG,TITLE(18)
      DOUBLE PRECISION A(40),ACHG(4C,4),DLD,ICDCT
      DOUBLE PRECISION      AID(20),CS(20),C(20,20),V(20),BB(20)
      DOUBLE PRECISION                 TST,     SSS,STDV,SPST
      DOUBLE PRECISION AMEAN,CD,AKNT,SOUT,SIGN,PCT,STER,AST
      DOUBLE PRECISION BETA,BVAR,BSTD,TRAT,SIZE,AM,DF,RR,R,ADJR,ADJRR
      REAL   DSAV(1C,101)
C
 1056 FORMAT (18A4)
 1000 FORMAT(6X,I2,1X,I2,1X,3(I1,1X),F9.0)
  510 FORMAT (6X,24A1)
 1001 FORMAT (6X,16A4)
 1057 FORMAT ('-',2X,'INPUT DATA FORMAT = ',2X,16A4    )
 1002 FORMAT  ('1', 2X,18A4)
 8001 FORMAT(6X,F2.C,1X,F2.0,1X,F2.0,1X,F6.0 )
 8002 FORMAT ('0',  2X,'ACHG(',I2,') = ',F3.0,1X,F3.0,1X,F3.0,1X,F6.0  )
  615 FORMAT('-',2X,'INPUT DATA',  4X,15(9A1,3X))
 9000 FORMAT (1H0,1CH C(I,J)=    ,6F6.2,//)
 9002 FORMAT(1H ,7H SIZE=  ,F10.2)
 1066 FORMAT ('-',2X,'VAR NO.',6X,'NAME',17X,'MEAN',3X,  'POP STD DEV',
     12X,   'SAMP STD DEV'  )
 1067 FORMAT (' ',I5,5X,2CA1, 3F12.5)
 1006 FORMAT (6X,10(F1.0,1X))
  488 FORMAT (44HCTOO FEW DEGREES OF FREEDOM, TRY ANCTHER EQN,/,14H SAMP
     1LE SIZE =,/,18HINDEP VAR IN EQN =,F5.0)
 2000 FORMAT (8H0VAR NO.,I3,20HLEADS TO SINGULARITY,5X8HVALUE IS,F12.8)
  446 FORMAT ('0'/,'0',2X, 'INTERCORRELATION  MATRIX'    )
  459 FORMAT (' ',35X,8I6,/,'0',24X,9I5)
  455 FORMAT (' ',6X,20A1,5X,I4,2X,16F6.2)
 4002 FORMAT ('2',I1,119('-') )
  660 FORMAT  ('0', 2X,18A4 )
 3009 FORMAT('-',18X,'DEPENDENT VARIABLE IS NUMBER  -',I3,' -',3X, 20A1)
  490 FORMAT('0'15X ,'SAMPLE SIZE =',F6.1,7X,'DEGREES FREEDOM ='F6.1  )
 3010 FORMAT('C',6X,'R**2 = ',F5.3,7X,'R = ',F5.3,8X,'STC ERR OF EST =',
     1F8.3 )
  472 FORMAT('0',2X,'ADJ R**2 = ',F5.3,3X,'ADJ R = ',F5.3,4X,'ADJ STD ER
     1R OF EST =' ,F8.3     )
  475 FORMAT('-', 2X,'LINEAR REGRESSION EQUATION CCEFFICIENTS'  )
 3012 FORMAT('0', 2X'VAR NO.'4X'NAME'19X6HREG WT,4X,'BTA WT',4X,'WT SIG'
     1,3X,'T-RATIO')
 3053 FORMAT (' ',2X , I4,3X, 21A1, F12.5,3F11.5 )
  469 FORMAT (1HC,///)
 3091 FORMAT('-',2X, 'RESULTS ARE IN RAW SCORES')
```

```
 3086 FORMAT ('C',2X,  'END OF PROBLEM'// '2')
  430 FORMAT('-', 2X,'END OF REGRESSION PGM',/'1')
      TOL=0.00005
   10 DO 14 I=1,20
      AID(I)=0.0
      DO 14 J=1,20
   14 C(I,J)=0.0
C   READ TITLE
      READ (5,1C56)(TITLE(I),I=1, 18)
C   READ NUMBER OF VARIABLES CARD , NVAR = 10 MAX
   16 READ (5,10CC ) NIND,NVAR,ICOR,IPRT,ICHG,TST
      NIND=NIND+1
      ICDCT=0
      NVARM=NVAR+1
      M=NVARM
      IF(NVARM .LT. 1) GOTO 850
   20 DO 24  I=2, NVARM
C   READ IN NAMES OF DATA VARIABLES CARD COLS 7 - 27.
   24 READ (5,51C)(ANAM(J,I),J=1,24)
C   READ IN DATA FORMAT
      READ (5,10C1)(FMT(I),I=1,16 )
C   WRITE TITLE OF OUTPUT DATA
      WRITE (6,1002)(TITLE(K),K=1,18)
C   WRITE FORMAT USED TO READ IN AND WRITE OUT THE PGM'S INPUT DATA
      WRITE(6,1057)(FMT(K),K=1,16 )
      IF(ICHG .LE. C) GOTO 38
      KCHG=0
      NTRANS = 4C
      DO 35    L=1, NTRANS
C   READ IN DESIRED VAR. TRANSFORMATIONS, MAX = 10.
      READ (5,80CI)(ACHG(L,J),J=1,4)
      IF(ACHG(L,1) .LE. 0) GOTO 38
C   WRITE OUT TRANSFCRMATICNS
      WRITE (6,8002) L,(ACHG(L,J),J=1,4)
      KCHG=KCHG+1
   35 CONTINUE
C   38 WRITE(6,8C03)
C   WRITE NAMES CF INPUT DATA VARIABLES.
   38 WRITE(6,615) ((ANAM(JK,IJ), JK=1,9),IJ=2,NIND)
C. READ IN NIND   COLS OF DATA PER DATA FORMAT CARD,  REF. STMT 1001
   40 READ (5,FMT)(A(I),I=2,NIND)
      IF(A(2) .GE. TST) GOTO 60
C   WRITE LIST OF INDEPENDENT VARIABLE INPUT DATA.
      WRITE (6,FMT)(A(I),I=2,NIND)
      A(1)=1.0
      ICDCT=ICDCT+1
      IF(ICHG .GT. C) CALL CHANGG
      DO 55 I=1,NVARM
      DO 55 J=I,NVARM
   55 C(I,J)=C(I,J)+A(I)*A(J)
      GOTO 40
   60 SIZE=C(1,1)
C
C WRITE ROW OF ** IF A(2) .GE. TST, PROVIDED  LAST DATA CARD HAS ALL 9'S
C     OTHERWISE THE LAST DATA VARIABLE IS LISTED IN THE CUTPUT TWICE.
      WRITE (6,FMT)(A(I),I=2,NIND)
      V(1)=DSQRT(C(1,1))
      DO 74   I=2,NVARM
      IF (C(I,I) .LE. 0) GOTO  72
```

```
      V(I)=(SIZE*C(I,I)-C(1,I)*C(1,I))/(SIZE*SIZE)
      IF (V(I) .LE. 0) GOTO  72
      V(I)=DSQRT(V(I))*V(1)
      GOTO 74
   72 V(I)=1.0*V(1)
   74 CONTINUE
      SSS=DSQRT(SIZE)
      SSS=1.0/SSS
C  WRITE DATA STATISTICS HEADING
      WRITE(6,1066)
      DO 76 I=2,NVARM
      STDV=V(I)*SSS
      SPST=V(I)*SSS
      SPST=SPST*SPST
      CD=(SIZE*SIZE)/((SIZE-1.0)*(SIZE-1.0))
      SPST=SPST*CD
      SPST=DSQRT(SPST)
      AMEAN=C(1,I)/SIZE
      K=I-1
C  WRITE DATA STATISTICS
      WRITE (6,1067)K,(ANAM(J,I), J=1,20),AMEAN,STDV,SPST
   76 CONTINUE
C     READ MODEL DEFINATION CARDS (INVERT CARDS)
   80 READ (5,1006)(AI(I),I=2,NVARM)
      AI(1)=1.0
      IF (AI(2)-3.0) 82,800,850
   82 AKNT=0.0
      DO  84 I=1,NVARM
      IF(A(I) .NE. 1.0) GOTO  84
      AKNT=AKNT+1.0
   84 CONTINUE
      IF (SIZE .GT. AKNT) GOTO  86
C
      WRITE (6,488)SIZE,AKNT
      GOTO 80
   86 DO  92  I=1,NVARM
      IF (AID(I) .GT. 0) GOTO 90
      CS(I)=1.0/V(I)
      GOTO 92
   90 CS(I)=V(I)
   92 CONTINUE
      DO 94   I=1,NVARM
      DO   94 J=I,NVARM
   94 C(I,J)=C(I,J)*CS(I)*CS(J)
      DO 180 N=1,M
      IF (AI(N) .EQ. 1) GOTO 96
      SOUT=0.0
      GOTO 98
   96 SOUT=1.0
   98 IF (SOUT .EQ. AID(N)) GOTO 180
      IF(C(N,N) .LE. 0) GOTO 170
      IF (C(N,N) .LE. TCL) GOTO 170
  101 C(N,N)=1.0/C(N,N)
      IF (SOUT .GT. 0) GOTO 106
  102 DO 105 I=1,M
      IF (I-N)103,105,104
  103 C(I,N)=(1.0-2.0*AID(I))*C(I,N)
      GO TO 105
  104 C(N,I)=(1.0-2.0*AID(I))*C(N,I)
```

```
  105 CONTINUE
  106 DO 111 I=1,M
      DO 111 J=I,M
      SIGN=1.0-2.0*DABS(SOUT-AID(I)*AID(J))
      IF (I-N)107,111,110
  107 IF (J-N)108,111,109
  108 C(I,J)=C(I,J)+SIGN*C(I,N)*C(J,N)*C(N,N)
      GO TO 111
  109 C(I,J)=C(I,J)+SIGN*C(I,N)*C(N,J)*C(N,N)
      GO TO 111
  110 C(I,J)=C(I,J)+SIGN*C(N,I)*C(N,J)*C(N,N)
  111 CONTINUE
      IF (SOUT .LE. C) GOTO 116
  112 DO 115 I=1,M
      SIGN=1.0-2.0*AID(I)
      IF (I-N)113,115,114
  113 C(I,N)=SIGN*C(I,N)*C(N,N)
      GO TO 115
  114 C(N,I)=SIGN*C(N,I)*C(N,N)
  115 CONTINUE
      GO TO  169
  116 DO  168 I=1,M
      IF (I-N) 161, 168, 162
  161 C(I,N)=C(I,N)*C(N,N)
      GO TO  168
  162 C(N,I)=C(N,I)*C(N,N)
  168 CONTINUE
  169 AID(N)=SOUT
      GOTO 180
  170 K=N-1
C
      WRITE (6,200C)K,C(N,N)
      AI(N)=2.0
      GOTO 190
  180 CONTINUE
  190 DO 250  I=1,NVARM
      IF (AID(I))25C,241,242
  241 CS(I)=V(I)
      GOTO 250
  242 CS(I)=1.0/V(I)
  250 CONTINUE
      IF(ICOR .LE. 0) GOTO 444
C  WRITE TITLE 'INTERCORRELATION MATRIX'
      WRITE (6,446)
      MKB=M-1
C
      WRITE (6,459)(IJ,IJ=1,MKB)
      DO 447 II=1,M
      DO 447 JJ=II,M
  447 C(JJ,II)=C(II,JJ)
      DO 450 II=2,M
      MK=II-1
C  WRITE INTERCORRELATION MATRIX DATA.
      WRITE (6,455)(ANAM(JN,II),JN=1,20),MK,(C(II,JJ),JJ=2,II)
  450 CONTINUE
      DO 458 II=1,M
      DO 458 JJ=1,M
  458 C(II,JJ)=C(II,JJ)*CS(II)*CS(JJ)
      ICOR=0
```

```
      GOTO 80
  444 DO 254  I=1,NVARM
      DO 254  J=I,NVARM
  254 C(I,J)=C(I,J)*CS(I)*CS(J)
      AM=0.0
      DO 299  I=1,NVARM
      IF (AID(I)-1.0)299,280,299
  280 AM=AM+1.0
  299 CONTINUE
      DO 300  I=2,NVARM
      IF (AI(I) .LE. 1.0) GOTO 300
  301 PCT=C(I,I)/(V(I)*V(I))
      STER=C(I,I)/SIZE
      AST=SIZE/(SIZE-AM)
      ADSSR=STER*AST
      STER=DSQRT(STER)
      ADSSR=SQRT(ADSSR)
      DF=SIZE-AM
      RR=1.0-PCT
      R = OSQRT(RR)
      ADJRR=1.0-(1.0-RR)*((SIZE-1.0)/(SIZE-AM))
      IF (ADJRR .GT. 0.0) GOTO 340
      ADJRR=0.0
      ADJR=0.0
      GOTO 344
  340 ADJR=DSQRT(ADJRR)
  344 K=I-1
C  WRITE REGRESSION EQUATION STATISTICS
      WRITE (6,4002)IPRT
      WRITE (6, 660) (TITLE(K), K=1,18)
      WRITE (6,3009)K,(ANAM(MK,I),MK=1,20)
      WRITE (6,490)SIZE,OF
      WRITE (6,3010)RR,R,STER
      WRITE (6,472)AOJRR,AOJR,AOSSR
      WRITE(6,475)
C  WRITE TITLES FOR REGRESSION EQUATION COEFFICIENT TABLE.
      WRITE (6,3012)
      KJ=0
      DO  320 J=1,NVARM
      IF (AIO(J) .NE. 1.0) GOTO 320
      IF (I-J)324,320,325
  324 BETA=C(I,J)
      GOTO 326
  325 BETA=C(J,I)
  326 BVAR=C(I,I)*C(J,J)/(SIZE-AM)
      BSTD = OSQRT(BVAR)
      TRAT=BETA/BSTO
      K=J-1
      KJ=KJ+1
      KZ(KJ)=K
      BB(KJ)=BETA
      BVAR=BETA*V(J)/V(I)
C  WRITE REGRESSION EQUATION COEFFICIENTS
      WRITE (6,3053)K,(ANAM(NJ,J),NJ=1,21),BETA,BVAR,BSTD,TRAT
  320 CONTINUE
  300 CONTINUE
C
      WRITE (6,3091)
      GOTO 80
```

```
C
  800 WRIJE (6,3086)
      IF(NVARM .LE. 0) GOTO 850
  820 GOTO 10
  850 WRITE(6,430)
      STOP
      END
      BLOCK  DATA
      COMMON   A,ACHG,CLD,ICDCT
      COMMON AI(40),ANAM(25,40),FMT(18),KZ(20),KCHG,TITLE(18)
      DOUBLE PRECISION A(40),ACHG(4C,4),DLD,ICDCT
      DATA ANAM/'C','O','N','S','T','A','N','T',' ','T','E','R','M',
     1 987*' '/
      END
      SUBROUTINE CHANGG
      COMMON   A,ACHG,CLD,ICDCT
      COMMON AI(40),ANAM(25,40),FMT(18),KZ(20),KCHG,TITLE(18)
      DOUBLE PRECISION A(40),ACHG(40,4),DLD,ICDCT
C
      DO 100 I=1,KCHG
      KBAB=ACHG(I,1)
      ND=IDINT( ACHG(I,2) )
      MD=IDINT( ACHG(I,3) )
      LD=IDINT( ACHG(I,4) )
      MD=MD+1
      ND=ND+1
      LD=LD+1
      DLD=ACHG(I,4)
      GO TO (1,2,3,4,5,6,7,8,9,10,11,12),KBAB
    1 A(MD)=A(NC)**CLD
      GO TO 100
    2 A(MD)=1.0/(A(ND)**DLC)
      GO TO 100
    3 A(MD)=A(NC)*A(LD)
      GO TO 100
    4 A(MD)=A(NC)/A(LC)
      GO TO 100
    5 A(MD)=A(NC)-A(LD)
      GO TO 100
    6 A(MD)=DSQRT(A(ND))
      GO TO 100
    7 A(MD)=DLOG(A(ND))
      GO TO 100
    8 A(MD)=DEXP(A(NC))
      GO TO 100
    9 A(MD)=A(ND)+A(LC)
      GO TO 100
   10 A(MC)=ICDCT
      GO TO 100
   11 A(MD)=A(NC)+CLD
      GOTO 100
   12 A(MD)=A(NC)*(DLD)
  100 CONTINUE
      RETURN
      ENC
```

2 TYPICAL OUTPUT

C3 MILLIAMPERE LINEAR REGRESSION PGM, C3 IC-NOMINAL DATA, VCE = 2.0 VOLT

INPUT DATA FORMAT = (16X,3F10.3)

ACHG(1) = 7. 1. 4. C.0

ACHG(2) = 3. 4. 5. 4.

INPUT DATA

COLLECTCR	BETA	VBE
0.106	0.0	0.658
C.540	108.375	0.702
1.080	108.00C	0.720
1.293	106.500	0.725
1.616	107.667	0.730
2.140	104.800	0.738
2.658	103.6CC	0.745
3.183	105.00C	0.749
3.688	101.00C	0.754
4.192	100.800	0.758
5.213	102.100	0.763
6.176	96.300	0.770
7.207	103.10C	0.772
8.230	1C2.30C	0.775

****

INPUT DATA STATISTICS

VAR NO.	NAME	MEAN	POP STD DEV	SAMP STD DEV
1	COLLECTOR CURRENT	3.38018	2.44564	2.63377
2	BETA	96.39586	26.92444	28.99555
3	VBE	0.73993	0.03071	0.03307
4	LN(IC, MILLIAMPS)	C.79623	1.12882	1.21565
5	LN(IC, MILLIAMPS)**2	1.90821	1.64586	1.77246

INTERCORRELATICN MATRIX

		1	2	3	4	5
COLLECTOR CURRENT	1	1.00				
BETA	2	0.29	1.C0			
VBE	3	0.86	C.67	1.00		
LN(IC, MILLIAMPS)	4	0.86	0.68	1.00	1.00	
LN(IC, MILLIAMPS)**2	5	C.59	-0.59	0.13	C.12	1.00

C3 MILLIAMPERE LINEAR REGRESSION PGM, C3 IC-NOMINAL DATA, VCE = 2.0 VOLT

DEPENDENT VARIABLE IS NUMBER - 18 - BETA

SAMPLE SIZE = 14.0 DEGREES FREEDOM = 11.0

R**2 = 0.918 R = 0.958 STD ERR OF EST = 7.705

ADJ R**2 = 0.903 ADJ R = 0.950 ADJ STD ERR OF EST = 8.692

LINEAR REGRESSION EQUATION COEFFICIENTS

VAR NO.	NAME	REG WT	BTA WT	WT SIG	T-RATIO
0	CONSTANT TERM	103.31701	3.83729	3.79670	27.21229
4	LN(IC, MILLIAMPS)	18.08139	0.75807	2.07263	8.72391
5	LN(IC, MILLIAMPS)**2	-11.17180	-0.68292	1.42151	-7.85908

DEPENDENT VARIABLE IS NUMBER - 18 - VBE

SAMPLE SIZE = 14.0 DEGREES FREEDOM = 11.0

R**2 = 0.999 R = 1.000 STD ERR OF EST = 0.001

ADJ R**2 = 0.999 ADJ R = 1.000 ADJ STD ERR OF EST = 0.001

LINEAR REGRESSION EQUATION COEFFICIENTS

VAR NO.	NAME	REG WT	BTA WT	WT SIG	T-RATIO
0	CONSTANT TERM	0.71791	23.37929	0.00038	1879.57800
4	LN(IC, MILLIAMPS)	0.02716	0.99830	0.00021	130.24236
5	LN(IC, MILLIAMPS)**2	0.00021	0.01114	0.00014	1.45294

RESULTS ARE IN RAW SCORES

END OF PROBLEM

C3 MICROAMPERE LINEAR REGRESSION PGM, C3 IC-NOMINAL DATA, VCE = 2.0 VOLT

INPUT DATA FORMAT = (16X,3F10.3)

ACHG(1) = 7. 1. 4. C.0

ACHG(2) = 3. 4. 5. 4.

INPUT CATA

COLLECTCR	BETA	VBE
106.500	0.0	0.658
540.COO	108.375	C.702
1080.000	108.00C	0.720
1292.999	106.500	0.725
1615.999	107.667	C.730
2139.999	1C4.800	0.738
2658.000	1C3.60C	0.745
3183.000	105.00C	0.749
3688.000	101.00C	0.754
4191.996	100.800	0.758
5212.996	102.100	0.763
6175.996	96.30C	0.770
7206.996	103.10C	0.772
8229.996	102.30C	0.775

****

INPUT CATA STATISTICS

VAR NO.	NAME	MEAN	POP STD DEV	SAMP STD DEV
1	COLLECTCR CLRRENT	338C.17693	2445.64334	2633.76975
2	BETA	96.39586	26.92444	28.99555
3	VBE	C.73993	0.03071	0.03307
4	LN(IC, MICROAMPS)	7.70398	1.12882	1.21565
5	LN(IC, MICROAMPS)**2	60.62560	15.87491	17.09606

INTERCORRELATICN MATRIX

		1	2	3	4	5
COLLECTOR CURRENT	1	1.00				
BETA	2	0.29	1.00			
VBE	3	0.86	0.67	1.00		
LN(IC, MICROAMPS)	4	0.86	0.68	1.00	1.00	
LN(IC, MICROAMPS)**2	5	0.90	0.60	1.00	0.99	1.00

C3 MICROAMPERE LINEAR REGRESSION PGM, C3 IC-NOMINAL DATA, VCE = 2.0 VOLT

DEPENDENT VARIABLE IS NUMBER - 18 - BETA

SAMPLE SIZE = 14.0 DEGREES FREEDOM = 11.0

R**2 = 0.918 R = 0.958 STD ERR OF EST = 7.705

ADJ R**2 = 0.903 ADJ R = 0.950 ADJ STD ERR OF EST = 8.692

LINEAR REGRESSION EQUATION COEFFICIENTS

VAR NO.	NAME	REG WT	BTA WT	WT SIG	T-RATIO
0	CONSTANT TERM	-554.67056	-20.60101	68.90291	-8.05003
4	LN(IC, MICROAMPS)	172.42553	7.22899	19.99124	8.62505
5	LN(IC, MICROAMPS)**2	-11.17180	-6.58700	1.42151	-7.85908

DEPENDENT VARIABLE IS NUMBER - 18 - VBE

SAMPLE SIZE = 14.0 DEGREES FREEDOM = 11.0

R**2 = 0.999 R = 1.000 STD ERR OF EST = 0.001

ADJ R**2 = 0.999 ADJ R = 1.000 ADJ STD ERR OF EST = 0.001

LINEAR REGRESSION EQUATION COEFFICIENTS

VAR NO.	NAME	REG WT	BTA WT	WT SIG	T-RATIO
0	CONSTANT TERM	0.54023	17.59311	0.00693	77.93624
4	LN(IC, MICROAMPS)	0.02429	0.89277	0.00201	12.07571
5	LN(IC, MICROAMPS)**2	0.00021	0.10742	0.00014	1.45295

RESULTS ARE IN RAW SCORES

```
3  INSTRUCTIONS FOR PREPARING LINEAR REGRESSION PROGRAM
    CONTROL CARDS

C       1.TITLE CARD WITH FORMAT = 18A4, MAY START IN COL 1, ALL OTHER
C          CONTROL CARDS MUST START IN CD COL 7.
C
C       2.MAIN PARAMETER CD - PROVIDES I/O PARAMETERS TO PGM.
C
C         A. COL 7-8 NIND = I2, NUMBER OF INDEPENDENT INPUT VARIABLES
C
C         B. COL 10-11 NVAR =NIND + NO. VAR CREATED BY TRANSFORMATIONS
C
C         C. COL 13 ICOR = I1, IF ICOR=1, THEN INTERCORRELATION MATRIX
C            APPEARS IN THE OUTPUT, OTHERWISE NO MATRIX IS PRINTED.
C            IF NVAR .GE. 15 SET ICOR = 0. THIS RESTRICTION DUE TO FMT.
C            BESURE TO INCLUDE A BLANK CARD AS THE FIRST INVERT CARD
C            WHENEVER ASKING FOR THE INTERCORRELATION MATRIX.
C
C         D.COL 15 IPRT=CARRIAGE CONTROL,SETS PAGE SPACING BETWEEN EQNS
C
C                 0  MEANS FOR PRINTER TO SKIP TWO LINES
C                 -  MEANS FOR PRINTER TO SKIP THREE LINES
C                 1  MEANS FOR PRINTER TO SKIP  TO NEW PAGE
C                 2  MEANS FOR PRINTER TO SKIP TO NEW HALF PAGE
C
C         E. COL 17 ICHG=1 TO TRANSFORM VARIABLES, FOR NO TRANS, ICHG=0
C
C         F. COL 19-27  TST = F FORMAT NO. .GE. MAX. DATA VAR + 0.5
C              TST USED BY PGM TO SIGNAL END OF DATA.
C
C       3. DATA VAR NAME CDS, ONE NAME PER CD. FORMAT IS 6A4.
C
C       4. INPUT DATA FORMAT CD, EX.(16X,6F10.3) = ENTRY NEEDED TO READ
C            6 SETS OF INPUT DATA. PARENTHESES ARE REQUIRED IN ENTRY.
C            IF A DIFFERENT FORMAT DESIRED CHECK INTO DATA TITLES OF
C            FORMAT STMT NO. 615.
C
C       5. CHANGE CARDS DEFINE THE VARIABLE TRANSFORMATIONS FOR FORMING
C              LINEAR REGRESSIONS EQUATIONS AS DEFINED BY THE INVERT CDS
C
C         A. TRANSFORMATION CARD INPUT CODES ARE...
C
C                      I  COLS 7-8     IN FIXED POINT FORMAT ONLY
C                      J  COLS 10-11   IN FIXED POINT FORMAT ONLY
C                      K  COLS 13-14   IN FIXED POINT FORMAT ONLY
C                 L OR DL COLS 19-27   IN  FLOATING POINT FORMAT ONLY
```

```
C
C         B.  TRANSFORMATICNS OPERATIONS CODE IS..
C                  01  A(K)=A(J)**DL
C                  02  A(K)=1./(A(J)**DL)
C                  03  A(K)=A(J)*A(L)
C                  C4  A(K)=A(J)/A(L)
C                  05  A(K)=A(J)-A(L)
C                  06  A(K)=SQRT(A(J))
C                  07  A(K)=ELOG(A(J))
C                  08  A(K)=EXPF(A(J))
C                  C9  A(K)=A(J)+A(J))
C                  10  A(K)=ICDT
C                  11  A(K)=A(J)+DL
C                  12 A(K)=A(J)*(DL)
C
C         C. AN EXAMPLE TRANSFORMATICN SPECIFICATICN CODE IS...
C
C            INDEX        II JJ KK LLLLLL
C            CARD COL    7890123456789
C            TRANS CODE C4 03 07 06.
C
C            THIS CODE INSTRUCTS THE PGM THAT DESIRED TRANSFORMATION
C            IS 04  A(K)=A(J)/A(L). PGM IS TO DIVIDE VAR 3 BY VAR 6 AND
C            STORE THE RESULT IN VAR A(7).
C
C         D. NOTE  DL IS USED OCCASIONALLY AS A FIXED POINT CONSTANT
C            MULTIPLIER OR EXPONENT. DL IS ENTERED IN COLS 8-12  IN
C            PLACE OF L.. NOTE THAT L AND DL ARE NEVER REQUIRED IN THE
C            SAME TRANSFCRMATION.
C
C         E. ENTRY 00 IN COLS 7-8 MEANS MC MORE CHANGE CARDS PRESENT.
C
C        7. AN ENC OF DATA CARD IS PUT AT THE CONCLUSION OF EACH SET.
C            THIS END CARD MUST HAVE A VALUE (IN THE FIELD OF THE FIRST
C            VARIABLE) THAT IS GREATER THAN TST BY 0.1 OR MORE,  NOTE
C            VALUE FOR TST WAS ENTERED ON THE MAIN PARAMETER CARD 2.
C            SEVERAL SETS OF DATA CAN BE CONCATENATED IN ONE CMPTR RUN.
C
C        8. INVERT CARCS DEFINE MCDELS USING COLS 7 - 37,WITH COL 7 FOR
C            VAR(1).
C
C         A. AN EXAMPLE INVERT CARC CODE IS....
C            CARD COL 7 8 9 0 1 2 3
C            ENTRY    2 0 0 1 1 C C
C
C         B. REGRESSICN MODEL CODES....
C
C            COL  7=C = VARIABLE NOT USED
C                   1 = INDEPENDENT VARIABLE
C                   2 = DEPENDENT VARIABLE
C                   3 = RESTART PGM WITH ALL NEW VARIABLES, CAUSES
C            PGM START THRU THE SET OF MODELS ONCE AGAIN AS DEFINED BY
C            THE INVERT CD USING NEW DATA.
C            COL  7=4 = END CF PGM CARD, TERMINATES ALL PROGRAM CALC.
C                   3 = RESTART PGM WITH ALL NEW VARIABLES
C
C
```

9. SAMPLE DATA DECK

```
C1 MICROAMPERE LINEAR REGRESSION PGM, C1 TRANSISTCR DATA, VCE = 2.0 VOLTS
C      03 05 1 2 1 12000.0                  NIND,NVAR,ICOR,IPRT,ICHG,TST
C      COLLECTOR CURRENT                    VARIABLE NAME CARD
C      BETA
C      VBE
C      LN(IC, MICROAMPS)
C      LN(IC, MICROAMPS)**2
C      (16X,3F10.3)                         INPUT DATA FCRMAT CD
C      07 01 04 0.0                         VARIABLE TRANSFORMATION CD
C      03 04 05 4.C
C      00
C 3.00  C.00025C    38.3C0    90.667    0.632    2.368    0.000    C1DAT 99
C 3.00  0.000500    62.100    95.200    0.652    2.348    0.000    C1DAT100
C 3.00  0.C01000   112.3CC   100.40C    0.672    2.328    0.000    C1DAT101
C 3.00  0.002000   213.5CC   101.200    0.691    2.307    0.002    C1DAT102
C 3.C0  0.C03000   32C.800   107.30C    0.703    2.297    0.000    C1DAT103
C 3.00  0.C04000   428.30C   107.50C    0.710    2.290    0.000    C1DAT104
C 3.00  0.C0500C   535.30C   107.000    0.716    2.281    0.003    C1DAT105
C 3.00  0.C0600C   642.4CC   107.100    0.720    2.279    0.001    C1DAT106
C 3.00  0.007000   748.80C   106.400    0.724    2.274    0.002    C1DAT107
C 3.00  C.008000   854.8CC   106.C00    0.728    2.272    C.000    C1DAT108
C 3.C0  0.C10000  1C79.000   112.10C    0.734    2.266    0.000    C1DAT109
C 3.00  C.01200C  1287.999   104.50C    0.737    2.263    0.000    C1DAT110
C 3.00  0.01500C  1589.0C0   100.334    0.746    2.254    0.000    C1DAT111
C 3.00  C.C20000  2113.0CC   1C4.800    0.755    2.245    0.000    C1DAT112
C 3.00  0.025000  262C.000   101.400    0.762    2.238    C.000    C1DAT113
C 3.00  C.030000  3117.99S    99.60C    0.767    2.232    0.001    C1DAT114
C 3.00  0.C3500C  36C8.000    98.00C    0.772    2.226    0.002    C1DAT115
C 3.C0  0.C4000C  4C94.999    97.40C    0.776    2.224    0.000    C1DAT116
C 3.00  C.050000  5054.996    96.C0C    0.782    2.217    0.001    C1DAT117
C 3.00  C.C60000  5S2S.996    87.50C    0.790    2.210    0.000    C1DAT118
C 3.00  0.C70000  6864.996    93.50C    0.794    2.206    C.000    C1DAT119
C 3.00  0.080000  7666.9S6    80.20C    C.800    2.200    C.000    C1DAT120
C 3.00  0 000100    24.700     0.0      0.603    2.397    0 000    C1DAT 98
C9999999S9999999999999999S9S9S9S9999999TST CARD TO SIGNAL END CF DATA
C                                           COMPLETELY BLANK CD SINCE ICOR=1
C      C 2 0 1 1                            REGRESSION MCDEL OEFN -INVERT CD
C      0 0 2 1 1
C      4                                    END CF PRCGRAM CARD

C          *****************ENO CF SAMPLE DATA DECK**********************
C
```

4 TYPICAL INPUT DATA DECK

```
C3 MILLIAMPERE LINEAR REGRESSION PGM, C3 TRANSISTOR DATA, VCE = 2.0 VOLTS
      03 05 1 2 1 12000.0
      COLLECTOR CURRENT
      BETA
      VBE
      LN(IC, MILLIAMPS)
      LN(IC, MILLIAMPS)**2
      (16X,3F10.3)
      07 01 04 0.0
      03 04 05 4.0
      00
  2.00  0.001000  0.106500    0.0       0.658     1.342     0.0         C3DAT 27
  2.00  0.005000  0.540000  108.375     0.702     1.298     0.000       C3DAT 28
  2.00  0.010000  1.080000  108.000     0.720     1.280     0.000       C3DAT 29
  2.00  0.012000  1.292999  106.500     0.725     1.275     0.000       C3DAT 30
  2.00  0.015000  1.615999  107.667     0.730     1.270     0.000       C3DAT 31
  2.00  0.020000  2.139999  104.800     0.738     1.262     0.000       C3DAT 32
  2.00  0.025000  2.658000  103.600     0.745     1.255     0.000       C3DAT 33
  2.00  0.030000  3.183000  105.000     0.749     1.251     0.000       C3DAT 34
  2.00  0.035000  3.688000  101.000     0.754     1.246     0.000       C3DAT 35
  2.00  0.040000  4.191999  100.800     0.758     1.242     0.000       C3DAT 36
  2.00  0.050000  5.212999  102.100     0.763     1.237     0.000       C3DAT 37
  2.00  0.060000  6.176000   96.300     0.770     1.230     0.000       C3DAT 38
  2.00  0.070000  7.207000  103.100     0.772     1.228     0.000       C3DAT 39
  2.00  0.080000  8.230000  102.300     0.775     1.225     0.000       C3DAT 40
99999999999999999999999999999999999999999999999999999999999999999999999999999999

      0 2 0 1 1
      0 0 2 1 1
      3
```

```
C3 MICROAMPERE LINEAR REGRESSION PGM, C3 TRANSISTOR DATA, VCE = 2.0 VOLTS
      03 05 1 2 1 12000.0
      COLLECTOR CURRENT
      BETA
      VBE
      LN(IC, MICROAMPS)
      LN(IC, MICROAMPS)**2
      (16X,3F10.3)
      07 C1 04 0.0
      03 C4 C5 4.C
      00
  2.00  C.C01C0C    1C6.500     0.0        0.658     1.342    0.0        C3DAT 27
  2.00  0.005000    54C.000   108.375      0.702     1.298    0.000      C3DAT 28
  2.00  0.C10000   108C.0CC   108.00C      0.720     1.280    0.000      C3DAT 29
  2.00  0.C12000   1292.999   106.500      0.725     1.275    0.000      C3DAT 30
  2.00  0.C1500C   1615.999   107.667      0.730     1.270    0.000      C3DAT 31
  2.00  0.020000   2139.999   104.800      0.738     1.262    0.000      C3DAT 32
  2.00  0.C25000   2658.0CC   103.60C      0.745     1.255    0.000      C3DAT 33
  2.00  0.C30000   3183.0C0   105.00C      0.749     1.251    0.000      C3DAT 34
  2.00  0.035000   3688.000   101.000      0.754     1.246    0.000      C3DAT 35
  2.00  0.040000   4191.996   100.80C      0.758     1.242    0.000      C3DAT 36
  2.00  0.050000   5212.996   102.10C      0.763     1.237    0.000      C3DAT 37
  2.00  C.060000   6175.996    96.300      C.770     1.230    C.000      C3DAT 38
  2.00  0.070000   7206.996   103.10C      0.772     1.228    0.000      C3DAT 39
  2.00  C.080000   8229.996   102.300      0.775     1.225    C.000      C3DAT 40
99999999999999999999999999999999999999999999999999999999999999999999999999999999

      C 2 0 1 1
      0 C 2 1 1
      5
```

```
      5  LINEAR REGRESSION INPUT DATA TRANSFORMATION PROGRAM

         THIS PROGRAM CALCULATES, USING MEASURED CARD INPUT DATA, BETA,
         MEAN COLLECTOR CURRENT,ICM, DATA FORMAT CONVERSION,
         AND PUNCHES CARDS WHICH ARE THEN USED AS INPUT DATA TO LINEAR
         REGRESSION MODEL COEFICIENT CALCULATING PROGRAM.

      COMMON VCE(26C),IB(260),IC(260),VCB(260),VBE(26C)
      COMMON BETA(260),TVCE(260),VCEDEV(260)
      COMMON DIC(260),DIB(26C),ICA(26C),ICM(260)
  200 FORMAT(' ',4X,F9.3,1X,F10.6,1X,F10.5,1X,F6.3,5X,F6.3)
  300 FORMAT('1')
  301 FORMAT('2',2X,'VCE',6X,'IB',5X,'IC-MEAN',5X,'BETA',7X,'VBE',7X,
     1   'VCB',6X,'VCEDEV'//)
C 400 FORMAT(' ',F5.2,2F10.6,F10.3,3F10.3,6X,'C1DAT',I3)               MA.REGSN
  400 FORMAT(' ',F5.2, F10.6,F10.3,F10.3,3F10.3,6X,'C1DAT',I3)         MICR.REG
C 401 FORMAT(' ',F5.2    ,2F10.6,F10.3,3F10.3,6X,'C1DAT',I3,5X,3F12.3)  MA.REGSN
  401 FORMAT(' ',F5.2,1F10.6,F10.3,F10.3,3F10.3,6X,'C1DAT',I3,5X,3F12.3)MICR.REG
  402 FORMAT('-',22X'********** END OF DATA ***********')
      REAL IB,IC,ICA,ICM
      N=000
      BETA(1) = C.CCC
      DIB(1) = 0.
      DIC(1) = 0.
      VBE(1) = 0.C
      IC(1) = C.C00C001
      IB(1) = C.CCCC0C1
      VCE(1) = C.5
      WRITE(6,3CC)
      DO 13 J=2,261
   12 N=N+1
      BETA(J) = C.000
   10 READ(5,20C,ENC=14)VCE(J),IB(J),IC(J),VBE(J),VCB(J)
   13 CONTINUE
   14 DO 16 J=2, N
      IF(VCE(J+1) .NE. VCE(J)) GOTO 15
    8 DIB(J+1) = IB(J+1) - IB(J)
      DIC(J+1) = IC(J+1) - IC(J)
      IF(IB(J+1) .EQ. IB(J)) GOTO 15
      ICM(J+1) = (IC(J+1) + IC(J))/2.C
      ICM(J+1) =  ICM(J+1)*1000.0
      BETA(J+1) = DIC(J+1)/DIB(J+1)
   15 TVCE(J)=VBE(J) + VCB(J)
      VCEDEV(J) = VCE(J) - TVCE(J)
      ICA(J) = ( IC(J) )/1.0
   16 CONTINUE
      VCE(N+1) = 1CC.C
   30 DO 40  J=2, N
      K=J-1
      IF(VCE(J+1) .GT. VCE(J)) WRITE(6,301)
      WRITE(6,401)VCE(J),IB(J),ICM(J),BETA(J), VBE(J),VCB(J),VCEDEV(J),K
     1,BETA(J  ),DIC(J  ),DIB(J  )
      WRITE(7,400)VCE(J),IB(J),ICM(J),BETA(J),VBE(J),VCB(J),VCEDEV(J),K
   40 CONTINUE
      WRITE(6,4C2)
      STOP
      END
```

APPENDIX D

REGRESSION MODEL EFFECTIVENESS PROGRAM

1 REGRESSION MODEL EFFECTIVENESS PROGRAM LISTING

```
      COMMON  VCE(200), BETAM(200),VBEM(200),BETAC(200),VBEC(200)
      COMMON  VDIFF(200),BDIFF(200),B(5),V(5),K,M,TITLE(8),UNITS(5)
      COMMON  INT
      REAL  ICM(200)
  102 FORMAT(6X,8A4)
  104 FORMAT(6X,4F12.5,21X,5A1)
  106 FORMAT(6X,4F12.5,21X,5A1)
  108 FORMAT(I1,F5.2,10X,3F10.3)
  200 FORMAT('1', 5X,'ANALYSIS OF REGRESSION EQUATION FOR BETA - '8A4/
     1'-' ,5X,'BETA = ',F10.5,' + ',F10.5,' * LN (IC) + ',F10.5,' * '
     2'(LN(IC))**2',/  )
  202 FORMAT('0',7X,'VCE',11X,'IC',10X,'MEASURED',6X,'CALCULATED'8X,
     1 'BETA',/,36X,'BETA',11X,'BETA',8X,'MEAS-CALC' ,/,
     2 7X,'VOLTS',6X,'(',5A1,'AMPS)',7X,'(--)',11X,'(--)',11X,'(--)'    )
  204 FORMAT(6X,F6.2,4F15.5)
  205 FORMAT('1', 5X,'ANALYSIS OF REGRESSION EQUATION FOR VBE - ',8A4,/
     1'-',5X,'VBE = ',F9.5,' + ',F9.5,' * LN (IC) + ',  F9.5,    ' * '
     2'(LN(IC))**2',/ )
  206 FORMAT('0',7X,'VCE',11X,'IC',10X,'MEASURED',6X,'CALCULATED'8X,
     I 'VBE',/,37X,'VBE ',10X,' VBE',8X,'MEAS-CALC',/ ,
     2 7X,'VOLTS',6X,'(',5A1,'AMPS)',7X,'(--)',11X,'(--)',11X,'(--)'    )
  208 FORMAT(6X,F6.2,4F15.5)
   10 K=0
      M=1
      READ(5,102,END=40)   (TITLE(II), II=1,8)
      READ(5,104) B(1),B(2),B(3),B(4),(UNITS(KK), KK=1,5)
      READ(5,106) V(1),V(2),V(3),V(4),(UNITS(KK), KK=1,5)
      B(4) = 0.0
      V(4) = 0.0
      DO 15 I=1,200
      READ(5,108,END=20) INT,VCE(I),ICM(I),BETAM(I),VBEM(I)
      IF(INT .GE. M) GOTO 20
      K = K + 1
   15 CONTINUE
   20 DO 30 J=1,K
      BETAC(J) = B(1)  + B(2)*(ALOG(ICM(J))) + B(3)*((ALOG(ICM(J)))**2)
       VBEC(J)  = V(1) + V(2)*(ALOG(ICM(J))) + V(3)*((ALOG(ICM(J)))**2)
      BDIFF(J) = BETAM(J) - BETAC(J)
      VDIFF(J) = VBEM(J) - VBEC(J)
   30 CONTINUE
      WRITE(6,200) (TITLE(JJ),JJ=1,8), B(1),B(2),B(3)
      WRITE(6,202)  (UNITS(KK), KK=1,5)
      WRITE(6,204) (VCE(I),ICM(I),BETAM(I),BETAC(I),BDIFF(I), I=1,K)
      WRITE(6,205) (TITLE(JJ),JJ=1,8), V(1),V(2),V(3)
      WRITE(6,206)  (UNITS(KK), KK=1,5)
      WRITE(6,208) (VCE(I),ICM(I),VBEM(I),VBEC(I),VDIFF(I), I=1,K)
      INT = 0
      GOTO 10
   40 STOP
      END
```

2 TYPICAL OUTPUT

ANALYSIS OF REGRESSION EQUATION FOR BETA - C3 IC-NOMINAL VCE=2.0 VDC

BETA = 103.31700 + 18.08138 * LN (IC) + -11.17180 * (LN(IC))**2

VCE VOLTS	IC (MILLIAMPS)	MEASURED BETA (--)	CALCULATED BETA (--)	BETA MEAS-CALC (--)
0.0	C.1C600	0.0	6.46486	-6.46486
0.0	0.540C0	108.37500	87.93373	20.44127
0.0	1.08000	1C8.00000	104.64238	3.35762
0.0	1.29300	106.50000	107.22557	-0.72557
0.0	1.61600	107.66699	109.42172	-1.75473
0.0	2.14000	104.79999	110.60689	-5.80690
0.0	2.65800	103.59999	110.31653	-6.71654
0.0	3.18300	1C5.00000	109.27560	-4.27560
0.0	3.68800	101.00000	107.88643	-6.88643
0.0	4.19200	10C.79999	106.28398	-5.48399
0.0	5.21300	102.C9999	102.71432	-0.61433
0.0	6.17600	96.29999	99.20450	-2.90451
0.0	7.20700	103.09999	95.44937	7.65062
0.0	8.23000	102.29999	91.79503	10.50496

ANALYSIS OF REGRESSION EQUATICN FOR VBE - C3 IC-NOMINAL VCE=2.0 VDC

VBE = 0.71791 + 0.02716 * LN (IC) + 0.00021 * (LN(IC))**2

VCE VCLTS	IC (MILLIAMPS)	MEASURED VBE (--)	CALCULATED VBE (--)	VBE MEAS-CALC (--)
0.0	0.10600	C.65800	0.65801	-0.00001
0.0	C.54000	C.70200	0.70125	0.00075
0.0	1.C8000	0.72000	0.72000	-0.00000
0.0	1.29300	0.72500	0.72490	0.00010
0.0	1.616C0	C.73000	0.73099	-0.00099
0.0	2.140C0	0.73800	0.73869	-0.00069
0.0	2.65800	0.74500	0.74466	0.00034
0.0	3.183C0	0.74900	0.74964	-0.00064
0.0	3.688C0	0.75400	0.75371	0.00029
0.0	4.19200	0.7580C	0.75727	0.00073
0.0	5.21300	C.76300	0.76333	-0.00033
0.0	6.17600	C.77000	0.76806	0.00194
0.0	7.207C0	C.77200	0.77237	-0.00037
0.0	8.23000	C.77500	0.77609	-0.00109

ANALYSIS OF REGRESSION EQUATION FOR BETA - C3 IC-NOMINAL VCE=2.0 VDC

BETA = -554.67041 + 172.42562 * LN (IC) + -11.17180 * (LN(IC))**2

VCE VOLTS	IC (MICROAMPS)	MEASURED BETA (--)	CALCULATED BETA (--)	BETA MEAS-CALC (--)
0.0	106.50000	0.0	6.78583	-6.78583
0.0	540.00000	108.37500	87.93408	20.44092
0.0	1080.00000	108.00000	104.64282	3.35718
0.0	1292.99878	106.50000	107.22583	-0.72583
0.0	1615.99878	107.66699	109.42187	-1.75488
0.0	2139.99878	104.79999	110.60718	-5.80719
0.0	2658.00000	103.59999	110.31689	-6.71690
0.0	3183.00000	105.00000	109.27612	-4.27612
0.0	3688.00000	101.00000	107.88672	-6.88672
0.0	4191.99219	100.79999	106.28442	-5.48444
0.0	5212.99219	102.09999	102.71484	-0.61485
0.0	6175.99219	96.29999	99.20483	-2.90485
0.0	7206.99219	103.09999	95.44971	7.65028
0.0	8229.99219	102.29999	91.79541	10.50458

ANALYSIS OF REGRESSION EQUATION FOR VBE - C3 IC-NOMINAL VCE=2.0 VDC

VBE = 0.54023 + 0.02429 * LN (IC) + 0.00021 * (LN(IC))**2

VCE VOLTS	IC (MICROAMPS)	MEASURED VBE (--)	CALCULATED VBE (--)	VBE MEAS-CALC (--)
0.0	106.50000	0.65800	0.65820	-0.00020
0.0	540.00000	0.70200	0.70136	0.00064
0.0	1080.00000	0.72000	0.72013	-0.00013
0.0	1292.99878	0.72500	0.72504	-0.00004
0.0	1615.99878	0.73000	0.73114	-0.00114
0.0	2139.99878	0.73800	0.73885	-0.00085
0.0	2658.00000	0.74500	0.74482	0.00018
0.0	3183.00000	0.74900	0.74980	-0.00080
0.0	3688.00000	0.75400	0.75388	0.00012
0.0	4191.99219	0.75800	0.75744	0.00056
0.0	5212.99219	0.76300	0.76351	-0.00051
0.0	6175.99219	0.77000	0.76824	0.00176
0.0	7206.99219	0.77200	0.77256	-0.00056
0.0	8229.99219	0.77500	0.77629	-0.00129

3 PROGRAM COMMENTS

1. INPUT CARDS

A. TITLE CARD - READ IN TRANSISTOR TYPE, METHOD OF OBTAINING IC - WHETHER NOMINAL VALUE OR MEAN VALUE OF IC IS USED IN THE COLLECTOR CURRENT, IC, DATA, AND THE COLLECTOR TO EMITTER VOLTAGE, VCE.

B. BETA EQUATION CARD - COEFFICIENTS OF THE REGRESSION MODEL BETA EQUATION ARE READ IN BY FORMAT(6X,4F12.5,5A1). NUMERIC VALUES ARE OBTAINED FROM REGRESSION PROGRAM OUTPUT. THE LAST 5A1 CHARACTERS COMPRISE THE COLLECTOR CURRENT UNIT'S PREFIX OR MULTIPLYING FACTOR. THAT IS MICRO IMPLIES MICROAMPERE UNITS, MILLI IMPLIES MILLIAMPERES UNITS, AND 5 BLANKS SIMPLY MEAN THAT THE CURRENT UNITS ARE IN AMPERES. THE PREFIX READ IN IS PLACED IN THE OUTPUT UNDER THE IC HEADING.

C. VBE EQUATION CARD - COEFFICIENTS OF THE REGRESSION MODEL VBE EQUATION ARE READ IN BY FORMAT(6X,4F12.5,5A1). NUMERIC VALUES ARE OBTAINED FROM REGRESSION PROGRAM OUTPUT.

D. DATA CARDS - READ USING FORMAT(I1,F5.2,3F10.3). CARDS MAY BE OBTAINED FROM THE LINEAR REGRESSION'S PROGRAM'S PUNCHED OUTPUT DATA SET.

E. END OF DATA CARD - ANY INTEGER GREATER THAN 1 ENABLES PROGRAM TO BEGIN READING ANOTHER DATA SET. IF A /* CARD IS ENCOUNTERED, THEN CONTROL IS RETURNED TO SYSTEM MONITOR. INTEGER MUST BE PLACED IN COLUMN 1. PROGRAM WILL NOT WORK WITHOUT THIS CARD.

2. SAMPLE DATA DECK

```
  C1 (C-NOMINAL VCE=2.C VDC                                   MILLIAMPERES
  104.48077    -5.28263     -3.90085                            C1 2. B MILLI
  0.74633      C.02786      -0.00499                            C1 2. V MILLI
                  0.007       0.C          0.404
                  C.017      99.50C        0.604
                  C.029      84.200        0.631
                  C.052      92.120        0.651
                  0.101      96.880        0.669
                  0.198      97.70C        0.688
                  0.300     101.50C        0.699
                  0.403     102.900        0.708
                  C.808     101.700        0.727
                  1.012     1C2.000        0.734
                  6.538      90.10C        0.795
                  7.293      75.5CC        0.802
3 (A NO. .GR. 1 IN COL 1 PERMITS RUN OF ANOTHER DATA SET OR TERMINATES PGM RUN)
```

```
C1 IC-NOMINAL VCE=2.0 VDC                                                  MICROAMPERES
-45.16509   48.60955    -3.90085                                             C1 2. B MICRO
0.31567     0.09683     -0.00499                                             C1 2. V MICRO
               6.750      0.0         0.404
              16.700     99.500       0.604
              29.330     84.200       0.631
              52.360     92.120       0.651
             706.300    102.000       0.724
             808.000    101.700       0.727
            1011.999    102.000       0.734
            5636.996     83.500       0.792
            6537.996     90.100       0.795
            7292.996     75.500       0.802
3

              END OF SAMPLE DATA DECK
```

4 TYPICAL INPLT DATA CECK

```
C3 IC-NOMINAL VCE=2.0 VDC                                            MILLIAMPERES
103.31701    18.08139     -11.17180                                    C3 2. B MILLI
0.71791      0.02716       0.08021                                     C3 2. V MILLI
                 C.106       0.C          0.658
                 0.540     108.375        0.702
                 1.080     108.000        C.720
                 1.293     106.50C        0.725
                 1.616     107.667        0.730
                 2.140     1C4.800        0.738
                 2.658     1C3.60C        0.745
                 3.183     1C5.00C        0.749
                 3.688     101.00C        0.754
                 4.192     100.800        C.758
                 5.213     102.100        0.763
                 6.176      96.300        C.770
                 7.207     103.10C        0.772
                 8.230     1C2.300        0.775
3
C3 IC-NOMINAL VCE=2.0 VDC                                            MICROAMPERES
-554.67C56   172.42553    -11.1718                                     C3 2. B MICRO
C.54023      C.02429       0.00021                                     C3 2. V MICRO
               1C6.50C       0.C          0.658
               540.C0C     108.375        0.702
              1C80.000     108.C0C        0.720
              1292.999     106.50C        0.725
              1615.999     107.667        0.730
              2139.999     104.800        0.738
              2658.000     103.60C        0.745
              3183.00C     105.00C        0.749
              3688.00C     101.00C        0.754
              4191.996     1C0.80C        0.758
              5212.996     102.1CC        0.763
              6175.996      96.30C        C.770
              7206.996     103.10C        0.772
              8229.996     102.300        0.775
3
```

APPENDIX E

NLAP OUTPUT SUBROUTINE LISTING

```
      SUBROUTINE ECB25                                                  DC250000
      COMMON CCSAV(6,200),IDWORD(74)
      DOUBLE PRECISION X(200)                                           DC250010
      COMMON NMAX,NNODE,NTERMS,NUMBL,NUMBR,NUMBC,IRTN,NTRACE,NSWTCH,KTO,DC250020
     1 NPRINT(10)                                                       DC250030
      COMMON E(200),EMIN(200),EMAX(200),AMP(200),AMPMIN(200),AMPMAX(200)DC250040
      COMMON Y(200),YMIN(200),YMAX(200),NINIT(200),NFIN(200),MODE1(200) DC250050
      COMMON YTERM(200),YTERMH(200),YTERML(200),IROWT(200),ICOLT(200)   DC250060
      COMMON ERROR1,ISEQ,MSEQ,MO,NUMMO,VFIRST(50),VSECND(50),VLAST(50)  DC250070
      COMMON MOBRN(50),MOPARM(50),MOSTEP(50),IWCOUT(4)                  DC250080
C                                                                       DC250090
C     THE FOLLOWING VARIABLES ARE USED ONLY IN THE ECAP D.C. ANALYSIS   DC250100
C                                                                       DC250110
      COMMON AX1,SMLEP(50),CURR(200),SMLE(200),EQUCUR(50),EX(200)       DC250120
      COMMON EB(200),AMPX(200),AMPB(200),VNOM(50),STDSQ(50),L,M,ITOL    DC250130
      COMMON JX1,JX4,JX5,DELTA,DUM1(28)                                 DC250140
C                                                                       DC250150
      COMMON LANG1(217),NTR,LANG2(47)
C                                                                       DC250170
      COMMON MATA(200,4,3),YX(200),YB(200),YTERMX(200),YTERMB(200)      DC250180
      COMMON WCMAX(50),WCMIN(50)                                        DC250190
      COMMON  CDSAV(2,200)
C                                                                       DC250200
      DOUBLE PRECISION SMLEP,CURR,SMLE,EQUCUR                           DC250210
C                                                                       DC250220
    1 IF(NTRACE)3,4,3                                                   DC250230
    2 FORMAT(6H ECB25 )
    3 WRITE(6,2)                                                        DC250250
C                                                                       DC250270
C     OUTPUT NODE VOLTAGES                                              DC250280
C                                                                       DC250290
    4 KNNODE = NNODE + 1                                                DC250292
      DO 5 I=1,NNODE
    5 CCSAV(6,I)=SMLEP(I)
      DO 6 K=KNNODE, NMAX                                               DC250293
    6 CCSAV(6,K) =  .0                                                  DC250294
C                                                                       DC250400
C     BRANCH VOLTAGES                                                   DC250410
C                                                                       DC250420
  101 DO 10 I=1,NMAX                                                    DC250430
      SMLE(I) = 0.0                                                     DC250440
      J=NINIT(I)                                                        DC250450
      IF(J)11,11,12                                                     DC250460
   12 SMLE(I) = SMLEP(J)                                                DC250470
   11 K=NFIN(I)                                                         DC250480
      IF ( K ) 10, 10, 14                                               DC250490
   14 SMLE(I) = SMLE(I) - SMLEP(K)                                      DC250500
```

```
   10 CONTINUE                                                      DC250510
      DO 16 I=1,NMAX
   16 CCSAV(1,I)=SMLE(I)
C                                                                   DC250630
C     ELEMENT VOLTAGES                                              DC250640
C                                                                   DC250650
      DO 17 I=1,NMAX
      SMLE(I)=SMLE(I)+EX(I)
   17 CCSAV(3,I)=SMLE(I)
C                                                                   DC250780
C     ELEMENT CURRENTS                                              DC250790
C                                                                   DC250800
  103 DO 20 I=1,NMAX                                                DC250810
   20 CURR(I)=YX(I)*SMLE(I)                                         DC250820
      IF(NTERMS)21,22,21                                            DC250830

   21 DO 23 I=1,NTERMS                                              DC250840
      NR=IROWT(I)                                                   DC250850
      NC=ICOLT(I)                                                   DC250860
      IF(SMLE(NC))24,23,24                                          DC250870
   24 CURR(NR)=CURR(NR)+YTERMX(I)*SMLE(NC)                          DC250880
   23 CONTINUE                                                      DC250890
   22 DO 27 I=1,NMAX
   27 CCSAV(4,I)=CURR(I)
C                                                                   DCB25092
C                                                                   DCB25093
C     BETA SAVE                                                     DCB25094
C                                                                   DCB25095
      DO 28  N = 1,NMAX                                             DCB25096
      NFBR = ICOLT(N)
      NTBR = IROWT(N)
      CDSAV(1,N) = 0.0
      CDSAV(2,N) = 0.0
      CDSAV(1,N) =  YTERM(N)
      IF(ICOLT(N) .EQ. 0) GOTO 28
      CDSAV(2,N) = (CDSAV(1,N))/(Y(NFBR))
  28  CONTINUE
C                                                                   DC251010
C     BRANCH POWER LOSSES                                           DC251020
C                                                                   DC251030
      DO 29 I=1,NMAX
   29 CCSAV(5,I)=CURR(I)*SMLE(I)
C                                                                   DC251140
C     BRANCH CURRENTS                                               DC251150
C                                                                   DC251160
  105 DO 30 I=1,NMAX                                                DC251170
   30 CURR(I)=CURR(I)-AMPX(I)                                       DC251180
C                                                                   DC251190
C     CHECK UNBALANCES                                              DC251200
C                                                                   DC251210
      DO 33 I=1,NNODE                                               DC251220
   33 X(I)=0.                                                       DC251230
      DO 36 I=1,NMAX                                                DC251240
      J=NINIT(I)                                                    DC251250
      K=NFIN(I)                                                     DC251260
      IF(K)34,34,35                                                 DC251270
   35 X(K)=X(K)+CURR(I)                                             DC251280
   34 IF(J)36,36,37                                                 DC251290
   37 X(J)=X(J)-CURR(I)                                             DC251300
```

```
 36 CONTINUE                                                            DC251310
    SUM=0.                                                              DC251320
    DO 38 I=1,NNODE                                                     DC251330
 38 SUM = SUM + DABS( X( I ))                                           DC251340
    IF(SUM-ERROR1)106,106,40                                            DC251350
 40 WRITE(6,141)                                                        DC251360
    WRITE (6,142)                                                       DC251370
141 FORMAT( // 43H SOLUTION NOT OBTAINED TO DESIRED TOLERANCE//)        DC251380
142 FORMAT( 6H NODES,15X,19H CURRENT UNBALANCES /)                      DC251390
    KMAX=NNODE                                                          DC251400
    IND=6                                                               DC251410
    GO TO 100                                                           DC251420
106 DO 42 I=1,NMAX
 42 CCSAV(2,I)=CURR(I)
    IF (JX4 .GT. 0) GOTO 900                                            DCA25104
398 GO TO (399,399,399,900),NTR                                         DCA25105
399 CALL BCDBN1(3,1,IDWORD(4) )
    IF(IDWORD(5) .GE. 1) IDWORD(73) = 0
    IDWORD(73) = IDWORD(73) + 1
C   IDWORD(73) = (IDWORD(73) + 1)/2
    WRITE(6,801) (IDWORD(IK), IK=7,72),IDWORD(3),IDWORD(4),IDWORD(73)
    WRITE (6,800)                                                       DCA25108
    IF (NMAX-NNODE) 400,500,401
400 MAX=NNODE
    MIN=NMAX
    GO TO 402
401 MAX=NMAX
    MIN=NNODE
402 DO 403 I=1,NMAX
403 X(I)=1.0/Y(I)
    WRITE (6,850) (I,E(I),AMP(I),X(I),(CCSAV(J,I),J=1,6),I,I=1,MIN)     DCB25117
    IF (MIN-MAX) 404,500,900
404 MIN=MIN+1
    WRITE (6,851)(I,E(I),AMP(I),X(I),(CCSAV(J,I), J=1,5), I= MIN,MAX)  DCB251
C   IF (IDWORD(74) .LE. 0) GOTO 900                          TEST       DCA25121
405 WRITE (8,852) (IDWORD(I), I=1, 73)                                  DCA25122
    WRITE( 7,852) (IDWORD(I), I=1, 73)
    WRITE (9,852) (IDWORD(I), I=1, 73)
    WRITE(10,852) (IDWORD(I), I=1, 73)
    WRITE(11,852) (IDWORD(I), I=1, 73)
    WRITE(7,853) (IDWORD(3),IDWORD(73),I,E(I),AMP(I),X(I),(CCSAV(J,I),
   1 J=1,6), I=1, MAX)
    WRITE(8,854) (IDWORD(3),IDWORD(73),I,E(I),AMP(I),X(I),(CCSAV(J,I),
   1 J=1,6), I=1, MAX)
    WRITE(9,855) (IDWORD(3),IDWORD(73),I,E(I),AMP(I),X(I),CDSAV(1,I),
   1CCSAV(1,I),CCSAV(2,I),CCSAV(5,I),CCSAV(6,I), I=1, MAX )
    WRITE(10,855) (IDWORD(3),IDWORD(73),I,E(I),AMP(I),X(I),CDSAV(2,I),
   1CCSAV(1,I),CCSAV(2,I),CCSAV(5,I),CCSAV(6,I), I=1, MAX )
    WRITE(11,855) (IDWORD(3),IDWORD(73),I,E(I),AMP(I),X(I),CDSAV(2,I),
   1CCSAV(1,I),CCSAV(2,I),CCSAV(5,I),CCSAV(6,I), I=1, MAX )
    GO TO 900
800 FORMAT (' BRANCH',3X,'VOLTAGE',5X,'CURRENT',7X,'BRANCH',7X,         DCB25121
   1'BRANCH',8X,'BRANCH',7X,'ELEMENT',7X,'ELEMENT',5X,'ELEMENT',7X      DCB25122
   2'NODE',6X,'NODE'/' NUMBER',3X,'SOURCE',6X,'SOURCE',6X,'RESISTANCE'DCB25123
   3,5X,'VOLTAGE',7X,'CURRENT',6X,'VOLTAGE',7X,'CURRENT',4X,            DCB25124
   4'POWER LOSS',4X,'VOLTAGE',3X,'NUMBER'/10X,'(VOLTS)',6X,'(MA)',9X,  DCB25125
   5'(OHMS)',7X,'(VOLTS)',8X,'(MA)',8X,'(VOLTS)',8X,'(MA)',7X,          DCB25126
   6'(WATTS)',6X,'(VOLTS)'///)                                          DCB25127
801 FORMAT('1',40X,66A1/'0',40X,'PROBLEM NO. =',2X,A1,A1,10X,'SOLUTION
```

```
      1 NO. = 'I3,////)
  850 FORMAT(' ',I4,2X,F10.3,   3PF12.4,3X,0PF12.2,2X,F11.6,2X,3PF12.6,
     12X,0PF11.6,2X,3PF12.6,2X,0PF10.4,2X,0PF11.6,2X,I5)
  851 FORMAT(' ',I4,2X,F10.3,   3PF12.4,3X,0PF12.2,2X,F11.6,2X,3PF12.6,
     12X,0PF11.6,2X,3PF12.6,2X,0PF10.4)
  852 FORMAT (1X,'C',1X,72A1,I3)
  853 FORMAT('P',A1,'-S',I3,'-B',I2,   F7.3,3PF8.2,0PF11.2,F7.3,3PF8.3,
     10PF7.3,3PF8.3,0PF6.3,0PF7.3)
  854 FORMAT(' P',A1,'-S',I3,'-B',I2,   F7.3,3PF6.2,0PF11.2,F7.3,3PF8.3,
     10PF7.3,3PF9.3,0PF6.3,0PF7.3)
  855 FORMAT(' P',A1,'-S',I3,'-B',I2,   F8.3,3PF8.2,0PF11.2, F9.3,F8.3,
     13PF9.3, 0PF7.3,CPF8.3)
  856 FORMAT(' P',A1,'-S',I3,'-B',I2,1X,F7.3,3PF8.2,0PF11.2,F7.3,3PF10.3
     1  ,0PF6.3,0PF7.3)
  860 FORMAT(' ',5X,'P',A1,'-S',I3,'-B',I2,5X,'ECB25',10X,'ICCLT(',I3,')
     1 =',I4,10X,'IROWT(',I3,') =',I4 )
C                                                                       DC251520
C     OUTPUT ROUTINE                                                    DC251530
C                                                                       DC251540
  100 LAST=0                                                            DC251550
  150 K=LAST+1                                                          DC251560
      LAST=LAST+4                                                       DC251570
      IF(LAST-KMAX)200,200,201                                          DC251580
  201 LAST=KMAX                                                         DC251590
  200 WRITE(6,203)K,LAST,(X(J),J=K,LAST)                                DC251600
  203 FORMAT(1X,I3,1H-,I3,2X,4(3X,E15.8))                               DC251610
      IF(KMAX-LAST)500,500,150                                          DC251620
C                                                                       DC251630
  500 GOTO 106
  900 RETURN                                                            DC251650
      END                                                               DC251660
```